EDITED BY

RUSLAN
MITKOV

The Oxford Handbook *of* COMPUTATIONAL LINGUISTICS

THE OXFORD HANDBOOK OF

COMPUTATIONAL LINGUISTICS

Edited by

RUSLAN MITKOV

OXFORD
UNIVERSITY PRESS

This book has been printed digitally and produced in a standard specification in order to ensure its continuing availability

OXFORD
UNIVERSITY PRESS

Great Clarendon Street, Oxford OX2 6DP

Oxford University Press is a department of the University of Oxford.
It furthers the University's objective of excellence in research, scholarship, and education by publishing worldwide in

Oxford New York

Auckland Cape Town Dar es Salaam Hong Kong Karachi
Kuala Lumpur Madrid Melbourne Mexico City Nairobi
New Delhi Shanghai Taipei Toronto

With offices in

Argentina Austria Brazil Chile Czech Republic France Greece
Guatemala Hungary Italy Japan South Korea Poland Portugal
Singapore Switzerland Thailand Turkey Ukraine Vietnam

Oxford is a registered trade mark of Oxford University Press in the UK and in certain other countries

Published in the United States
by Oxford University Press Inc., New York

Reprinted 2009

ISBN 978-0-19-927634-9

Contents

PART II PROCESSES, METHODS, AND RESOURCES

PART III APPLICATIONS

PREFACE

Computational Linguistics is an interdisciplinary field concerned with the processing of language by computers. Since machine translation began to emerge some fifty years ago (see Martin Kay's introduction below), Computational Linguistics has grown and developed exponentially. It has expanded theoretically through the development of computational and formal models of language. In the process it has vastly increased the range and usefulness of its applications. At a time of continuing and vigorous growth the *Oxford Handbook of Computational Linguistics* provides a much-needed reference and guide. It aims to be of use to everyone interested in the subject, including students wanting to familiarize themselves with its key areas, researchers in other areas in search of an appropriate model or method, and those already working in the field who want to discover the latest developments in areas adjacent to their own.

The *Handbook* is structured in three parts which reflect a natural progression from theory to applications.

Part I introduces the fundamentals: it considers, from a computational perspective, the main areas of linguistics such as phonology, morphology, lexicography, syntax, semantics, discourse, pragmatics, and dialogue. It also looks at central issues in mathematical linguistics such as formal grammars and languages, and complexity.

Part II is devoted to the basic stages, tasks, methods, and resources in and required for automatic language processing. It examines text segmentation, part-of-speech tagging, parsing, word–sense disambiguation, anaphora resolution, natural language generation, speech recognition, text-to-speech synthesis, finite state technology, statistical methods, machine learning, lexical knowledge acquisition, evaluation, sublanguages, controlled languages, corpora, ontologies, and tree-adjoining grammars.

Part III describes the main real-world applications based on Computational Linguistics techniques, including machine translation, information retrieval, information extraction, question answering, text summarization, term extraction, text data mining, natural language interfaces, spoken dialogue systems, multimodal/multimedia systems, computer-aided language learning, and multilingual on-line language processing.

Those who are relatively new to Computational Linguistics may find it helpful to familiarize themselves with the preliminaries in Part I before going on to subjects in Parts II and III. Reading Chapter 4 on syntax, for example, should help the reader to understand the account of parsing in Chapter 12.

To make the book as coherent and useful as possible I encouraged the authors to adopt a consistent structure and style of presentation. I also added numerous cross-references and, with the help of the authors, compiled a glossary. This latter will, I hope, be useful for students and others getting to know the field.

The diverse readership for whom the *Handbook* is intended includes university researchers, teachers, and students; researchers in industry; company directors; software engineers; computer scientists; linguists and language specialists; and translators. In sum the book is for all those who are drawn to this endlessly fascinating and rewarding field.

I thank all the contributors to the *Handbook* for the high quality of their input and their cooperation. I am particularly indebted to Eduard Hovy, Lauri Karttunen, John Hutchins, and Yorick Wilks for their helpful comments and to Patrick Hanks for his dedicated assistance in compiling the glossary. I thank John Davey, OUP's linguistics editor, for his help and encouragement. I acknowledge gratefully the support received from the University of Wolverhampton. Finally, I would like to express gratitude to my late mother Penka Georgieva Moldovanska for her kind words and moral support at the beginning of this challenging editorial project.

Ruslan Mitkov
March 2002

Abbreviations

ACL	Association for Computational Linguistics
AECMA	Association Européenne des Constructeurs de Matériel Aérospatiale (European Association of Aerospace Industries)
AFL	abstract family of languages
AI	artificial intelligence
ALPAC	Automatic Language Processing Advisory Committee
API	application programming interface
ATIS	Air Travel Information Systems
ATN	augmented transition network
AVM	attribute-value matrix
BNN	Broadcast News Navigator
CA	conversational analysis
CAI	computer-aided instruction
CALL	computer-assisted language learning
CART	classification and regressions tree
CAT	computer-aided (or computer-assisted) translation
CCG	Combinatory Categorial Grammar
CDNS	Columbia Digital News System
CF	context-free
CFG	context-free grammar
CG	Constraint Grammar
CIDE	*Cambridge International Dictionary of English*
CKY	Cocke–Kasami–Younger
CL	controlled language
CLAW	controlled language applications workshop
CLEF	Cross-Language Evaluation Forum
CLIR	cross-language information retrieval
CNF	Chomsky normal form
COBUILD	Collins Birmingham University International Language Database
COLLINS	*Collins English Dictionary*
COMPUTERM	International Workshop on Computational Terminology
COP	Constituent Object Parser
CP	cooperative principle

CS	context-sensitive
CSG	context-sensitive grammar
DAMSL	Dialog Act Markup in Several Layers
DARPA	Defense Advanced Research Projects Agency
DCG	definite clause grammar
DFA	deterministic finite automaton
DFM	Derivation Final Model
DLT	Distributed Language Translation (MT system)
DM	data mining
DNF	disjunctive normal form
DOBJ	direct object
DP	dynamic programming
DPDA	Deterministic Pushdown Dutomaton
DRI	Discourse Resource Initiative
DRS	discourse representation structure
DRT	Discourse Representation Theory
EAGLES	Expert Advisory Group on Language Engineering Standards
EBMT	example-based machine translation
EC	European Community
ECI	European Corpus Initiative
EDA	Exploratory Data Analysis
EDL	extended domain of locality
EDR	Electronic Dictionary Research (Institute)
ELDA	European Language Resources Distribution Agency
ELRA	European Language Resources Association
EM	Expectation Maximization
ENGCG	English Constraint Grammar
EPDA	embedded pushdown automaton
EUROPHRAS	European Society for Phraseology
FA	finite automaton
FACS	facial action coding system
FAPs	facial animation parameters
FAQ	frequently asked questions
FoG	Forecast Generator (Environment Canada's bilingual text generator for meteorological sublanguages)
FRD	factoring recursion from the domain of dependencies
FSA	finite-state automaton
FUF	Functional Unification Formalism
GNF	Greibach normal form
GPSG	Generalized Phrase Structure Grammar
GQ	generalized quantifier
HCRC	Human Communication Research Centre

HLT	Human Language Technology
HMM	hidden Markov model
HPKB	high performance knowledge bases
HPSG	Head-Driven Phrase Structure Grammar
ICALL	Intelligent Computer-Assisted Language Learning
ICT	Information and Communication Technology
IE	information extraction
IL	Intensional Logic
ILP	Inductive Logic Programming
IPA	International Phonetic Alphabet
IPFP	Iterative Proportional Fitting Procedure
IR	information retrieval
ISO	International Organization for Standardization
ISOBJ	'is the OBJect of'
IST	Information Society technology
IV	intransitive verb
KBMT	knowledge-based machine translation
KDD	Knowledge Discovery in Databases
KIF	knowledge interchange format
KNF	Kuroda normal form
KNN	K Nearest Neighbor
KPML	Komet-Penman MultiLingual resource development environment
KWIC	key-word in context
LCS	Lexical Conceptual Structure
LDC	Linguistic Data Consortium
LDOCE	*Longman Dictionary of Contemporary English*
LFG	Lexical Functional Grammar
LIN	linear grammar
LINDI	Linking Information for Novel Discovery and Insight
LLIN	left-linear grammar
LM	language model
LMI	linguistically motivated indexing
LPC	linear predictive coding
LREC	Language Resources and Evaluation Conference
LSI	Latent Semantic Indexing
LTAG	Lexicalized Tree-Adjoining Grammar
LTS	letter-to-sound
LVCSR	large vocabulary continuous speech recognition
MAHT	machine-aided human translation
MAP	maximum a posteriori
MATE	Multilevel Annotation Tools Engineering
MBROLA	multiband resynthesis overlap add

MC-LTAG	multi-component Lexicalized Tree-Adjoining Grammar
MCS	mildly context-sensitive language
MC-TAG	Multi-Component Tree-Adjoining Grammar
MDL	Minimal Description Length
ME	maximum entropy
MFC	Mel frequency cepstral
MFCC	Mel Frequency Cepstral Coefficients
ML	machine learning
MLLR	maximum likelihood linear regression
MMR	Maximum Marginal Relevance
MRD	machine-readable dictionary
MRL	meaning representation language
MT	machine translation
MTM	Meaning-Text Model
MUC	Message Understanding Conference
N	noun
NEC	Nippon Electric Company
NER	named entity recognition
NFA	non-deterministic finite automaton
NIML	non-indigenous minority language
NIST	National Institute of Standards and Technology (USA)
NL	natural language
NLG	natural language generation
NLI	natural language interface
NLP	natural language processing
NODE	*New Oxford Dictionary of English*
NP	noun phrase
NP	non-deterministic polynomial
NPDA	non-deterministic pushdown automaton
+Nsg	singular noun
OALD	*Oxford Advanced Learners' Dictionary*
OALDCE	*Oxford Advanced Learner's Dictionary of Current English*
OCR	Optical Character Recognition
OED	*Oxford English Dictionary*
OOV	out-of-vocabulary
OPP	optimum position policy
OT	Optimality Theory
P	probability
P	(deterministic) polynomial
PARC	Palo Alto Research Center
PaT-Nets	parallel transition networks
PCFG	Probabilistic Context-Free Grammar

PCM	Parallel Correspondence Model
PDA	pushdown automaton
PDF	probability density function
PLP	Perceptual Linear Prediction
PNF	Penttonen normal form
POS	part of speech
QA	question answering
RAGS	Reference Architecture for Generation Systems
RE	Recursively Enumerable language
REG	regular grammar
RLIN	right-linear grammar
RST	rhetorical structure theory
RTN	recursive transition network
S	sentence
SAT	speaker adaptive training
SBD	sentence boundary disambiguation
SDC	Systems Development Corporation
SDS	spoken dialogue system
SFG	systemic functional grammar
SGML	Standard Generalized Markup Language
SIGGEN	Special Interest Group on Generation
SIGPHON	ACL Special Interest Group in Computational Phonology
SL	source language
SLT	spoken language translation
SNOMED	Systematized Nomenclature of Medicine
Sp	speaker
SUBJ	subject
SURGE	Systemic Unification Realization Grammar for English
SUSY	Saarbrücker Übersetzungssystem (MT system)
TAG	tree-adjoining grammar
TDM	text data mining
TD-PSOLA	time-domain pitch-synchronous-overlap-add
TDT	topic detection and tracking
TEFL	Teaching English as a Foreign Language
TEI	Text-Encoding Initiative
TELRI	Trans-European Language Resources Infrastructure
TESL	Teaching English as a Second Language
TF*IDF	term frequency and inverse document frequency
TIDES	Translingual Information Detection, Extraction, and Summarization
TL	target language
TM	translation memory
TMR	text-meaning representation

TREC	Text Retrieval Conference
TRINDI	Task Oriented Instructional Dialogue
TSA	tree-structure analysis
TTS	text-to-speech
UMLS	Unified Medical Language System
URL	universal resource locator
+Vb	verb
VP	verb phrase
+VpastT	verb past participle
WFST	well-formed substring table
WOZ	Wizard of Oz
WSD	word-sense disambiguation
WSJ	*Wall Street Journal*
WWW	World Wide Web
XDOD	Xerox Document On Demand
XIFSP	Xerox Incremental Finite State Processing
XML	eXtensible Markup Language
XRCE	Xerox Research Centre Europe

INTRODUCTION

MARTIN KAY

Computational Linguistics is about as robust a field of intellectual endeavour as one could find, with its books, journals, conferences, professorial chairs, societies, associations and the like. But, of course, it was not always so. Computational Linguistics crept into existence shyly, almost furtively. When shall we say it all began? Perhaps in 1949, when Warren Weaver wrote his famous memorandum suggesting that translation by machine might be possible. The first conference on machine translation took place at MIT in 1952 and the first journal, *Mechanical Translation*, began in 1954. However, the phrase 'Computational Linguistics' started to appear only in the mid-1960s. The journal changed its name to *Mechanical Translation and Computational Linguistics* in 1965 but the words 'and Computational Linguistics' appeared in very small type. This change coincided with the adoption of the journal by the Association for Machine Translation and Computational Linguistics, which was formed in 1962.

The term 'Computational Linguistics' was probably coined by David Hays during the time that he was a member of the Automatic Language Processing Advisory Committee of the National Academy of Sciences. The publication of this committee's final report, generally known as the ALPAC report, certainly constituted one of the most dramatic moments in the history of the field—proposing, as it did, that machine translation be abandoned as a short-term engineering goal in favour of more fundamental scientific research in language and language processing. Hays saw this coming and realized that, if the money that had been flowing into machine translation could be diverted into a new field of enquiry, the most pressing requirement was for the field to be given a name. The name took hold. Redirection of the funds did not.

Progression from machine translation to Computational Linguistics occurred in 1974 when *Machine Translation and Computational Linguistics* was replaced by the *American Journal of Computational Linguistics*, which appeared initially only in microfiche form. In 1980, this became *Computational Linguistics*, which is still alive and vigorous today.

By the 1980s, machine translation began to look practical again, at least to some people and for some purposes and, in 1986, the circle was completed with the publication of the first issue of *Computers and Translation*, renamed *Machine Translation* in 1988. The *International Journal of Machine Translation* followed in 1991.

Warren Weaver's vision of machine translation came from his war-time experience as a cryptographer and he considered the problem to be one of treating textual material, by fundamentally statistical techniques. But the founders of Computational Linguistics were mostly linguists, not statisticians, and they saw the potential of the computer less in the possibility of deriving a characterization of the translation relation from emergent properties of parallel corpora, than in carrying out exactly, and with great speed, the minutely specified rules that they would write. Chomsky's *Syntactic Structures* (1957) served to solidify the notion of grammar as a deductive system which therefore seemed eminently suited to computer applications. The fact that Chomsky himself saw little value in such an enterprise, or that the particular scheme of axioms and rules that he advocated was ill suited to the automatic analysis of text, did nothing to diminish the attractiveness of the general idea.

Computational Linguistics thus came to be an exercise in creating and implementing the formal systems that were increasingly seen as constituting the core of linguistic theory. If any single event marks the birth of the field, it is surely the proposal by John Cocke in 1960 of the scheme for deriving all analyses of a string with a grammar of binary context-free rules that we now know as the Cocke–Kasami–Younger algorithm. It soon became clear that more powerful formalisms would be required to meet the specific needs of human language, and more general chart parsers, augmented transition networks, unification grammars, and many other formal and computational devices were created.

There were two principal motivations for this activity. One was theoretical and came from the growing perception that the pursuit of computational goals could give rise to important advances in linguistic theory. Requiring that a formal system be implementable helped to ensure its internal consistency and revealed its formal complexity properties. The results are to be seen most clearly in syntactic formalisms such as Generalized Phrase Structure Grammar, Lexical Functional Grammar, and Head Driven Phrase Structure as well as in application of finite-state methods to phonology and morphology.

The second motivation, which had existed from the beginning, came from the desire to create a technology, based on sound scientific principles, to support a large and expanding list of practical requirements for translation, information extraction, summarization, grammar checking, and the like. In none of these enterprises is success achievable by linguistic methods alone. To varying extents, each involves language not just as a formal system, but as a means of encoding and conveying information about something outside, something which, for want of a better term, we may loosely call 'the world'. Much of the robustness of language comes from the imprecision and ambiguity which allow people to use it in a casual manner. But this works only because people are able to restore missing information and resolve ambiguities on the basis of what makes sense in a larger context provided not only by the surrounding words but by the world outside. If there is any field that should be responsible for the construction of comprehensive, general models of the world, it

is presumably artificial intelligence, but the task is clearly a great deal more daunting even than building comprehensive linguistic models, and success has been limited.

As a result, Computational Linguistics has gained a reputation for not measuring up to the challenges of technology, and this in turn has given rise to much frustration and misunderstanding both within and outside the community of computational linguists. There is, of course, much that still remains to be done by computational linguists, but very little of the responsibility for the apparently poor showing of the field belongs to them. As I have said, a significant reason for this is the lack of a broader technological environment in which Computational Linguistics can thrive. Lacking an artificial intelligence in which to embed their technology, linguists have been forced to seek a surrogate, however imperfect, and many think they have found it in what is generally known as 'statistical natural language processing'.

Roughly speaking, statistical NLP associates probabilities with the alternatives encountered in the course of analysing an utterance or a text and accepts the most probable outcome as the correct one. In 'the boy saw the girl with the telescope', the phrase 'with the telescope' is more likely to modify 'saw' than 'the girl', let us say, because 'telescope' has often been observed in situations which, like this one, represent it as an instrument for seeing. This is an undeniable fact about seeing and telescopes, but it is not a fact about English. Not surprisingly, words that name phenomena that are closely related in the world, or our perception of it, frequently occur close to one another so that crisp facts about the world are reflected in somewhat fuzzier facts about texts.

There is much room for debate in this view. The more fundamentalist of its proponents claim that the only hope for constructing useful systems for processing natural language is to learn them entirely from primary data as children do. If the analogy is good, and if Chomsky is right, this implies that the systems must be strongly predisposed towards certain kinds of languages because the primary data provides no negative examples and the information that it contains occurs, in any case, in too weak dilution to support the construction of sufficiently robust models without strong initial constraints.

If, as I have suggested, text processing depends on knowledge of the world as well as knowledge of language, then the proponents of radical statistical NLP face a stronger challenge than Chomsky's language learner because they must also construct this knowledge of the world entirely on the basis of what they read about it, and in no way on the basis of direct experience. The question that remains wide open is: Just how much of the knowledge of these two kinds that is required for NLP is derivable, even in principle, from emergent properties of text? The work done over the next few years should do much to clarify the issue and thus to suggest the direction that the field will follow thereafter.

This book stands on its own in the sense that it will not only bring people working in the field up to date on what is going on in parallel specialities to their own, but also introduce outsiders to the aims, methods, and achievements of computational

linguists. The chapters of Part I have the same titles that one might expect to find in an introductory text on general linguistics. With the exception of the last, they correspond to the various levels of abstraction on which linguists work, from individual sounds to structures that span whole texts or dialogues, to the interface between meaning and the objective world, and the making of dictionaries. The difference, of course, is that they concentrate on the opportunities for computational exploration that each of these domains opens up, and on the problems that must be solved in each of them before they can contribute to the creation of linguistic technology.

I have suggested that requiring a formal system to be implementable led linguists to attend to the formal complexity properties of their theories. The last chapter of Part I provides an introduction to the mathematical notion of complexity and explores the crucial role that it plays in Computational Linguistics.

Part II of the book gives a chapter to each of the areas that have turned out to be the principal centres of activity in the field. For these purposes, Computational Linguistics is construed very broadly. On the one hand, it treats speech recognition and text-to-speech synthesis, the fundamentals of which are more often studied in departments of electrical engineering than linguistics and on the other, it contains a chapter entitled 'Corpora', an activity in which students of language use large collections of text or recorded speech as sources of evidence in their investigations. Part III is devoted to applications—starting, as is only fitting, with a pair of chapters on machine translation followed by a discussion of some topics that are at the centre of attention in the field at the present.

It is clear from the table of contents alone that, during the half century in which the field, if not the name, of Computational Linguistics has existed, it has come to cover a very wide territory, enriching virtually every part of theoretical linguistics with a computational and a technological component. However, it has been only poorly supplied with textbooks or comprehensive reference works. This book should go a long way towards meeting the second need.

PART I

FUNDAMENTALS

This page intentionally left blank

CHAPTER 1

PHONOLOGY

STEVEN BIRD

ABSTRACT

Phonology is the systematic study of the sounds used in language, and their composition into syllables, words, and phrases. **Computational phonology** is the application of formal and computational techniques to the representation and processing of phonological information. This chapter will present the fundamentals of descriptive phonology along with a brief overview of computational phonology.

1.1 PHONOLOGICAL CONTRAST, THE PHONEME, AND DISTINCTIVE FEATURES

There is no limit to the number of distinct sounds that can be produced by the human vocal apparatus. However, this infinite variety is harnessed by human languages into **sound systems** consisting of a few dozen language-specific categories, or **phonemes**. An example of an English phoneme is *t*. English has a variety of *t*-like sounds, such as the aspirated t^h of *ten*, the unreleased $t^\neg$ of *net*, and the flapped *ɾ* of *water* (in some dialects). In English, these distinctions are not used to differentiate words, and so we do not find pairs of English words which are identical but for their use of t^h versus $t^\neg$. (By comparison, in some other languages, such as Icelandic and Bengali, aspir-

ation is contrastive.) Nevertheless, since these sounds (or *phones*, or **segments**) are phonetically similar, and since they occur in *complementary distribution* (i.e. disjoint contexts) and cannot differentiate words in English, they are all said to be **allophones** of the English phoneme *t*.

Of course, setting up a few allophonic variants for each of a finite set of phonemes does not account for the infinite variety of sounds mentioned above. If one were to record multiple instances of the same utterance by the single speaker, many small variations could be observed in loudness, pitch, rate, vowel quality, and so on. These variations arise because speech is a motor activity involving coordination of many independent articulators, and perfect repetition of any utterance is simply impossible. Similar variations occur between different speakers, since one person's vocal apparatus is different from the next person's (and this is how we can distinguish people's voices). So 10 people saying *ten* 10 times each will produce 100 distinct acoustic records for the *t* sound. This diversity of tokens associated with a single type is sometimes referred to as *free variation*.

Above, the notion of phonetic similarity was used. The primary way to judge the similarity of phones is in terms of their *place* and *manner* of articulation. The consonant chart of the International Phonetic Alphabet (IPA) tabulates phones in this way,

	Bilabial	Labiodental	Dental	Alveolar	Postalveolar	Retroflex	Palatal	Velar	Uvular	Pharyngeal	Glottal
Plosive	p b			t d		ʈ ɖ	c ɟ	k g	q ɢ		ʔ
Nasal	m	ɱ		n		ɳ	ɲ	ŋ	ɴ		
Trill	ʙ			r					ʀ		
Tap or Flap				ɾ		ɽ					
Fricative	ɸ β	f v	θ ð	s z	ʃ ʒ	ʂ ʐ	ç ʝ	x ɣ	χ ʁ	ħ ʕ	h ɦ
Lateral fricative				ɬ ɮ							
Approximant		ʋ		ɹ		ɻ	j	ɰ			
Lateral approximant				l		ɭ	ʎ	ʟ			

Fig. 1.1 Pulmonic Consonants from the International Phonetic Alphabet

as shown in Fig. 1.1. The IPA provides symbols for all sounds that are contrastive in at least one language.

The major axes of this chart are for place of articulation (horizontal), which is the location in the oral cavity of the primary constriction, and manner of articulation (vertical), the nature and degree of that constriction. Many cells of the chart contain two consonants, one *voiced* and the other *unvoiced*. These complementary properties are usually expressed as opposite values of a *binary feature* [±voiced].

A more elaborate model of the similarity of phones is provided by the theory of **distinctive features**. Two phones are considered more similar to the extent that they agree on the value of their features. A set of distinctive features and their values for five different phones is shown in (1.1). (Note that many of the features have an extended technical definition, for which it is necessary to consult a textbook.)

(1.1)

	t	z	m	l	i
anterior	+	+	+	+	−
coronal	+	+	−	+	−
labial	−	−	+	−	−
distributed	−	−	−	−	−
consonantal	+	+	+	+	−
sonorant	−	−	+	+	+
voiced	−	+	+	+	+
approximant	−	−	−	+	+
continuant	−	+	−	+	+
lateral	−	−	−	+	−
nasal	−	−	+	−	−
strident	−	+	−	−	−

Statements about the distribution of phonological information, usually expressed with rules or constraints, often apply to particular subsets of phones. Instead of listing these sets, it is virtually always simpler to list two or three feature values which pick out the required set. For example [+labial, −continuant] picks out *b*, *p*, and *m*, shown in the top left corner of Fig. 1.1. Sets of phones which can be picked out in this way are called **natural classes**, and phonological analyses can be evaluated in terms of their reliance on natural classes. How can we express these analyses? The rest of this chapter discusses some key approaches to this question.

Unfortunately, as with any introductory chapter like this one, it will not be possible to cover many important topics of interest to phonologists, such as acquisition, diachrony, orthography, universals, sign language phonology, the phonology/syntax interface, systems of intonation and stress, and many others besides. However, numerous bibliographic references are supplied at the end of the chapter, and readers may wish to consult these other works.

1.2 Early Generative Phonology

Some key concepts of phonology are best introduced by way of simple examples involving real data. We begin with some data from Russian in (1.2). The example shows some nouns, in nominative and dative cases, transcribed using the International Phonetic Alphabet. Note that *x* is the symbol for a voiceless velar fricative (e.g. the *ch* of Scottish *loch*).

(1.2)

Nominative	*Dative*	*Gloss*
xlep	xlebu	'bread'
grop	grobu	'coffin'
sat	sadu	'garden'
prut	prudu	'pond'
rok	rogu	'horn'
ras	razu	'time'

Observe that the dative form involves suffixation of *-u*, and a change to the final consonant of the nominative form. In (1.2) we see four changes: *p* becomes *b*, *t* becomes *d*, *k* becomes *g*, and *s* becomes *z*.

Where they differ is in their *voicing*; for example, *b* is a *voiced* version of *p*, since *b* involves periodic vibration of the vocal folds, while *p* does not. The same applies to the other pairs of sounds. Now we see that the changes we observed in (1.2) are actually quite systematic. Such systematic patterns are called **alternations**, and this particular one is known as a **voicing alternation**. We can formulate this alternation using a *phonological rule* as follows:

(1.3) $\begin{bmatrix} \text{C} \\ \text{voiced} \end{bmatrix} \rightarrow [+\text{voiced}] \,/\, \_\_\, \text{V}$

A consonant becomes voiced in the presence of a following vowel

Rule (1.3) uses the format of early generative phonology. In this notation, C represents any consonant and V represents any vowel. The rule says that, if a voiceless consonant appears in the *phonological environment* '___ V' (i.e. preceding a vowel), then the consonant becomes voiced. By default, vowels have the feature [+voiced]), and so we can make the observation that the consonant *assimilates* the voicing feature of the following vowel.

One way to see if our analysis generalizes is to check for any nominative forms that end in a voiced consonant. We expect this consonant to stay the same in the dative form. However, it turns out that we do not find any nominative forms ending in a voiced consonant. Rather, we see the pattern in example (1.4). (Note that č is an alternative symbol for IPA ʧ.)

(1.4)

Nominative	*Dative*	*Gloss*
čerep	čerepu	'skull'
xolop	xolopu	'bondman'
trup	trupu	'corpse'
cvet	cvetu	'colour'
les	lesu	'forest'
porok	poroku	'vice'

For these words, the voiceless consonants of the nominative form are unchanged in the dative form, contrary to our rule (1.3). These cannot be treated as exceptions, since this second pattern is quite pervasive. A solution is to construct an artificial form which is the dative word form minus the *-u* suffix. We will call this the **underlying form** of the word. Example (1.5) illustrates this for two cases:

(1.5)

Underlying	*Nominative*	*Dative*	*Gloss*
prud	prut	prudu	'pond'
cvet	cvet	cvetu	'colour'

Now we can account for the dative form simply by suffixing the *-u*. We account for the nominative form with the following *devoicing rule*:

(1.6) $\begin{bmatrix} \text{C} \\ \text{+voiced} \end{bmatrix} \rightarrow [\ \text{voiced}] \ / \ \_\_\_ \ \#$

A consonant becomes devoiced word finally

This rule states that a voiced consonant is devoiced (i.e. [+voiced] becomes [–voiced]) if the consonant is followed by a word boundary (symbolized by #). It solves a problem with rule (1.3) which only accounts for half of the data. Rule (1.6) is called a *neutralization* rule, because the *voicing contrast* of the underlying form is removed in the nominative form. Now the analysis accounts for all the nominative and dative forms. Typically, rules like (1.6) can simultaneously employ several of the distinctive features from (1.1).

Observe that our analysis involves a certain degree of abstractness. We have constructed a new level of representation and drawn inferences about the underlying forms by inspecting the observed surface forms.

To conclude the development so far, we have seen a simple kind of phonological representation (namely sequences of alphabetic symbols, where each stands for a bundle of distinctive features), a distinction between levels of representation, and rules which account for the relationship between the representations on various levels. One way or another, most of phonology is concerned about these three things: representations, levels, and rules.

Finally, let us consider the plural forms shown in example (1.7). The plural morpheme is either *-a* or *-y*.

(1.7)

Singular	*Plural*	*Gloss*
xlep	xleba	'bread'
grop	groby	'coffin'
čerep	čerepa	'skull'
xolop	xolopy	'bondman'
trup	trupy	'corpse'
sat	sady	'garden'
prut	prudy	'pond'
cvet	cveta	'colour'
ras	razy	'time'
les	lesa	'forest'
rok	roga	'horn'
porok	poroky	'vice'

The phonological environment of the suffix provides us with no way of predicting which allomorph is chosen. One solution would be to enrich the underlying form once more (for example, we could include the plural suffix in the underlying form, and then have rules to delete it in all cases but the plural). A better approach in this case is to distinguish two *morphological classes*, one for nouns taking the *-y* plural, and one for nouns taking the *-a* plural. This information would then be an idiosyncratic property of each lexical item, and a morphological rule would be responsible for the choice between the *-y* and *-a* **allomorphs**. A full account of these data, then, must involve phonological, morphological, and lexical modules of a grammar.

As another example, let us consider the vowels of Turkish. These vowels are tabulated below, along with a decomposition into distinctive features: [high], [back], and [round]. The features [high] and [back] relate to the position of the tongue body in the oral cavity. The feature [round] relates to the rounding of the lips, as in the English *w* sound.[1]

(1.8)

	u	o	ü	ö	ı	a	i	e
high	+	−	+	−	+	−	+	−
back	+	+	−	−	+	+	−	−
round	+	+	+	+	−	−	−	−

Consider the following Turkish words, paying particular attention to the four versions of the possessive suffix. Note that similar data are discussed in Chapter 2.

(1.9)

ip	'rope'	ipin	'rope's'
kız	'girl'	kızın	'girl's'
yüz	'face'	yüzün	'face's'
pul	'stamp'	pulun	'stamp's'
el	'hand'	elin	'hand's'

[1] Note that there is a distinction made in the Turkish alphabet between the dotted *i* and the dotless *ı*. This *ı* is a high, back, unrounded vowel that does not occur in English.

çan	'bell'	çanın	'bell's'
köy	'village'	köyün	'village's'
son	'end'	sonun	'end's'

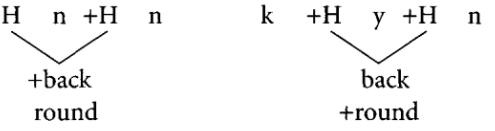
The possessive suffix has the forms *in*, *ın*, *ün*, and *un*. In terms of the distinctive feature chart in (1.8), we can observe that the suffix vowel is always [+high]. The other features of the suffix vowel are copied from the stem vowel. This copying is called **vowel harmony**. Let us see how this behaviour can be expressed using a phonological rule. To do this, we assume that the vowel of the possessive affix is only specified as [+high] and is underspecified for its other features. In the following rule, *C* denotes any consonant, and the Greek letter variables range over the + and – values of the feature.

(1.10) $\begin{bmatrix} V \\ +\text{high} \end{bmatrix} \rightarrow \begin{bmatrix} \alpha\text{back} \\ \beta\text{round} \end{bmatrix} / \begin{bmatrix} \alpha\text{back} \\ \beta\text{round} \end{bmatrix} \; C^*\_\_$

A high vowel assimilates to the backness and rounding of the preceding vowel

So long as the stem vowel is specified for the properties [high] and [back], this rule will make sure that they are copied onto the affix vowel. However, there is nothing in the rule formalism to stop the variables being used in inappropriate ways (e.g. α back → α round). So we can see that the rule formalism does not permit us to express the notion that certain features are shared by more than one segment. Instead, we would like to be able to represent the sharing explicitly, as follows, where ±H abbreviates [±high], an underspecified vowel position:

(1.11)



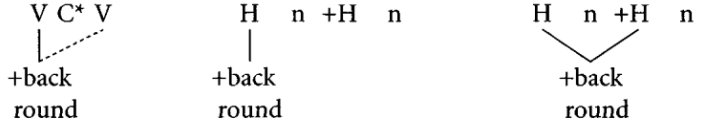
The lines of this diagram indicate that the backness and roundness properties are shared by both vowels in a word. A single vowel property (or type) is manifested on two separate vowels (tokens).

Entities like [+back, –round] that function over extended regions are often referred to as **prosodies**, and this kind of picture is sometimes called a *non-linear* representation. Many phonological models use non-linear representations of one sort or another. Here we shall consider one particular model, namely **autosegmental phonology**, since it is the most widely used non-linear model. The term comes from 'autonomous + segment', and refers to the autonomous nature of segments (or certain groups of features) once they have been liberated from one-dimensional strings.

1.3 Autosegmental Phonology

In autosegmental phonology, diagrams like those we saw above are known as *charts*. A chart consists of two or more *tiers*, along with some *association lines* drawn between the autosegments on those tiers. The *no-crossing constraint* is a stipulation that association lines are not allowed to cross, ensuring that association lines can be interpreted as asserting some kind of temporal overlap or inclusion. *Autosegmental rules* are procedures for converting one representation into another, by adding or removing association lines and autosegments. A rule for Turkish vowel harmony is shown below on the left in (1.12), where *V* denotes any vowel, and the dashed line indicates that a new association is created. This rule applies to the representation in the middle, to yield the one on the right.

(1.12)



In order to fully appreciate the power of autosegmental phonology, we will use it to analyse some data from an African tone language. Consider the data in Table 1.1. Twelve nouns are listed down the left side, and the isolation form and five contextual

Table 1.1 Tone Data from Chakosi (Ghana)

Word form	A. ___ isolation	B. i___ 'his…'	C. am goro ___ 'your (pl) brother's…'	D. ___ kũ 'one…'	E. am ___ wo dɔ 'your (pl)… is there'	F. jiine ___ ni 'that…'
1. bakɑ 'tree'	ˉˉ	ˉˉˉ	ˉˉˉˉ	ˉˉ	ˉˉˉ	ˉˉˉˉ
2. sɑkɑ 'comb'	ˉˇ	ˉˉˉ	ˉˉˉˇ	ˉˉ	ˉˉˉ	ˉˉˉˉ
3. buri 'duck'	ˉˉ	ˉˉˉ	ˉˉˉˉ	ˉˉ		ˉˉˉˉ
4. siri 'goat'	ˉˇ	ˉˉˇ	ˉˉˇ	ˉˉ	ˉˉˉ	ˉˉˉˉ
5. gɑdo 'bed'	ˉˉ	ˉˉˉ	ˉˉˉ	ˉˉ		ˉˉˉˉ
6. gɔrɔ 'brother'	ˉˉ	ˉˉˉ		ˉˉ	ˉˉˉ	
7. cɑ 'dog'	ˇ	ˉˉ	ˉˉˇ	ˉ	ˉˉ	ˉˉˉ
8. ni 'mother'	ˉ	ˉˉ	ˉˉˉ	ˉ		ˉˉˉ
9. jɔkɔrɔ 'chain'	ˉˉˉ	ˉˉˉˉ	ˉˉˉˉ	ˉˉˉ	ˉˉˉˉ	ˉˉˉˉ
10. tokoro 'window'	ˉˉˉ	ˉˉˉˉ	ˉˉˉˉ	ˉˉˉ	ˉˉˉˉ	ˉˉˉˉ
11. bulali 'iron'	ˉˉˉ	ˉˉˉˉ	ˉˉˉˉ	ˉˉˉ	ˉˉˉˉ	ˉˉˉˉ
12. misini 'needle'	ˉˉˉ	ˉˉˉˉ	ˉˉˉˉ	ˉˉˉ	ˉˉˉˉ	ˉˉˉˉ

forms are provided across the table. The line segments indicate voice pitch (the fundamental frequency of the voice); dotted lines are for the syllables of the context words, and full lines are for the syllables of the target word, as it is pronounced in this context. At first glace these data seem bewildering in their complexity. However, we will see how autosegmental analysis reveals the simple underlying structure of the data.

Looking across the table, observe that the contextual forms of a given noun are quite variable. For example *bulali* appears as ˉˉ_, ˉ_ˉ, _ˉˉ, and ˉˉˉ.

We could begin the analysis by identifying all the levels (here there are five), assigning a name or number to each, and looking for patterns. However, this approach does not capture the relative nature of tone, where _ˉ_ is not distinguished from ˉˉˉ. Instead, our approach just has to be sensitive to differences between adjacent tones. So these distinct tone sequences could be represented identically as +1, −2, since we go up a small amount from the first to the second tone (+1), and then down a larger amount −2. In autosegmental analysis, we treat *contour tones* as being made up of two or more *level tones* compressed into the space of a single syllable. Therefore, we can treat _˥ as another instance of +1, −2. Given our autosegmental perspective, a sequence of two or more identical tones corresponds to a single spread tone. This means that we can collapse sequences of like tones to a single tone.[2] When we retranscribe our data in this way, some interesting patterns emerge.

First, by observing the raw frequency of these intertone intervals, we see that −2 and +1 are by far the most common, occurring 63 and 39 times respectively. A −1 difference occurs 8 times, while a +2 difference is very rare (only occurring 3 times, and only in phrase-final contour tones). This patterning is characteristic of a *terrace tone language*. In analysing such a language, phonologists typically propose an inventory of just two tones, H (high) and L (low), where these might be represented featurally as [±hi]. In such a model, the tone sequence HL corresponds to ˉ_, a pitch difference of −2.

In terrace tone languages, an H tone does not achieve its former level after an L tone, so HLH is *phonetically realized* as ˉ_ˉ, (instead of ˉ_ˉ). This kind of H-lowering is called **automatic downstep**. A pitch difference of +1 corresponds to an LH tone sequence. With this model, we already account for the prevalence of the −2 and +1 intervals. What about −1 and +2?

As we will see later, the −1 difference arises when the middle tone of ˉ_ˉ (HLH) is deleted, leaving just ˉˉ. In this situation we write H!H, where the exclamation mark indicates the lowering of the following H due to a deleted (or *floating* low tone). This kind of H-lowering is called **conditioned downstep**. The rare +2 difference only occurs for an LH contour; we can assume that automatic downstep only applies when a LH sequence is linked to two separate syllables (_ˉ) and not when the sequence is linked to a single syllable (˯).

[2] This assumption cannot be maintained in more sophisticated approaches involving lexical and prosodic domains. However, it is a very useful simplifying assumption for the purposes of this presentation.

To summarize these conventions, we associate the pitch differences to tone sequences as shown in (1.13). Syllable boundaries are marked with a dot.

(1.13)	*Interval*	−2	−1	+1	+2
	Pitches	˗_	˗˗	_˗	˯
	Tones	H.L	H.!H	L.H	LH

Now we are in a position to provide tonal transcriptions for the forms in Table 1.1. Example (1.14) gives the transcriptions for the forms involving *bulali*. Tones corresponding to the noun are underlined.

(1.14) *Transcriptions of* bulali *'iron'*

bulali	'iron'	˗˗_	L.H.L
i bulali	'his iron'	˗˗˗_	H.H.!H.L
am goro bulali	'your (pl) brother's iron'	˗˗˗˗_	HL.L.L.L.H.L
bulali kũ	'one iron'	˗˗˗˗	L.H.H.L
am bulali wo dɔ	'your (pl) iron is there'	˗˗˗˗˗	HL.L.H.H.!H.L
jiine bulali ni	'that iron'	˗˗˗˗˗	L.H.H.!H.H.L

Looking down the right-hand column of (1.14) at the underlined tones, observe again the diversity of surface forms corresponding to the single lexical item. An autosegmental analysis is able to account for all this variation with a single spreading rule.

(1.15) *High tone spread*

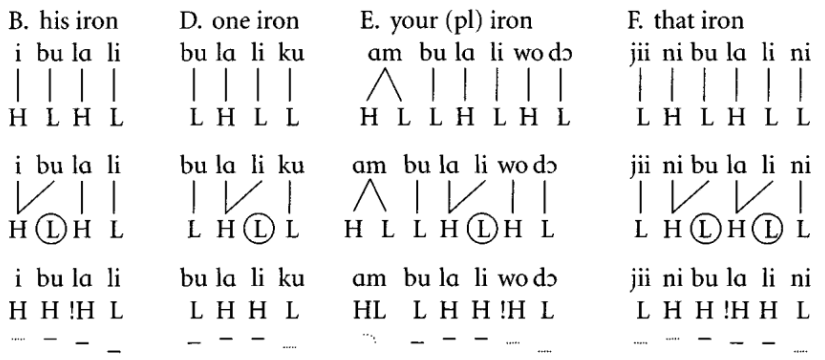
A high tone spreads to the following (non-final) syllable, delinking the low tone

Rule (1.15) applies to any sequence of three syllables (σ) where the first is linked to an H tone and the second is linked to an L tone. The rule spreads H to the right, delinking the L. Crucially, the L itself is not deleted, but remains as a *floating tone*, and continues to influence surface tone as downstep. Example (1.16) shows the application of the H spread rule to forms involving *bulali*. The first row of autosegmental diagrams shows the underlying forms, where *bulali* is assigned an LHL **tone melody**. In the second row, we see the result of applying H spread. Following standard practice, the floating low tones are circled. Where a floating L appears between two H tones, it gives rise to downstep. The final assignment of tones to syllables and the position of the downsteps are shown in the last row of the table.

Example (1.16) shows the power of autosegmental phonology—together with suitable underlying forms and appropriate principles of phonetic interpretation—in analysing complex patterns with simple rules. Space precludes a full analysis of the data; interested readers can try hypothesizing underlying forms for the other words, along with new rules, to account for the rest of the data in Table 1.1.

The preceding discussion of segmental and autosegmental phonology highlights the multi-linear organization of phonological representations, which derives from

(1.16)



the temporal nature of the speech stream. Phonological representations are also organized hierarchically. We already know that phonological information comprises words, and words, phrases. This is one kind of hierarchical organization of phonological information. But phonological analysis has also demonstrated the need for other kinds of hierarchy, such as the **prosodic hierarchy**, which builds structure involving syllables, feet, and intonational phrases above the segment level, and **feature geometry**, which involves hierarchical organization beneath the level of the segment. Phonological rules and constraints can refer to the prosodic hierarchy in order to account for the observed *distribution* of phonological information across the linear sequence of segments. Feature geometry serves the dual purpose of accounting for the inventory of contrastive sounds available to a language, and for the alternations we can observe. Here we will consider just one level of phonological hierarchy, namely the syllable.

1.4 Syllable Structure

Syllables are a fundamental organizational unit in phonology. In many languages, phonological alternations are sensitive to syllable structure. For example, *t* has several allophones in English, and the choice of allophone depends on phonological context. For example, in many English dialects, *t* is pronounced as the flap [ɾ] between vowels, as in *water*. Two other variants are shown in (1.17), where the phonetic transcription is given in brackets, and syllable boundaries are marked with a dot.

(1.17) a. atlas [æt^{ʔ}.ləs]
b. cactus [kæk.t^{h}əs]

Native English syllables cannot begin with *tl*, and so the *t* of *atlas* is syllabified with the preceding vowel. Syllable final *t* is regularly glottalized or unreleased in English, while syllable initial *t* is regularly aspirated. Thus we have a natural explanation for the patterning of these allophones in terms of syllable structure.

Other evidence for the syllable comes from loanwords. When words are borrowed into one language from another, they must be adjusted so as to conform to the legal sound patterns (or **phonotactics**) of the host language. For example, consider the following borrowings from English into Dschang, a language of Cameroon (Bird 1999).

(1.18) afruwa *flower*, akalatusi *eucalyptus*, alɛsa *razor*, alɔba *rubber*, aplɛŋgɛ *blanket*, asəkuu *school*, cɛɛn *chain*, dəək *debt*, kapinda *carpenter*, kɛsiŋ *kitchen*, kuum *comb*, laam *lamp*, lɛsi *rice*, luum *room*, mbasəku *bicycle*, mbrusi *brush*, mbərəək *brick*, mɛta *mat*, mɛtərasi *mattress*, ŋglasi *glass*, ɲjakasi *jackass*, mɛtisi *match*, nubatisi *rheumatism*, pɔkɛ *pocket*, ŋgalɛ *garden*, səsa *scissors*, tɛwɛlɛ *towel*, wasi *watch*, ziiŋ *zinc*

In Dschang, the **syllable canon** is much more restricted than in English. Consider the patterning of *t*. This segment is illegal in syllable-final position. In technical language, we would say that alveolars are not *licensed* in the syllable coda. In mɛta *mat*, a vowel is inserted, making the *t* into the initial segment of the next syllable. For dəək *debt*, the place of articulation of the *t* is changed to velar, making it a legal syllable-final consonant. For aplɛŋgɛ *blanket*, the final *t* is deleted. Many other adjustments can be seen in (1.18), and most of them can be explained with reference to syllable structure.

A third source of evidence for syllable structure comes from morphology. In Ulwa, a Nicaraguan language, the position of the possessive *infix* is sensitive to syllable structure. The Ulwa syllable canon is (C)V(V|C)(C), and any **intervocalic** consonant (i.e. consonant between two vowels) is syllabified with the following syllable, a universal principle known as **onset maximization**. Consider the Ulwa data in (1.19).

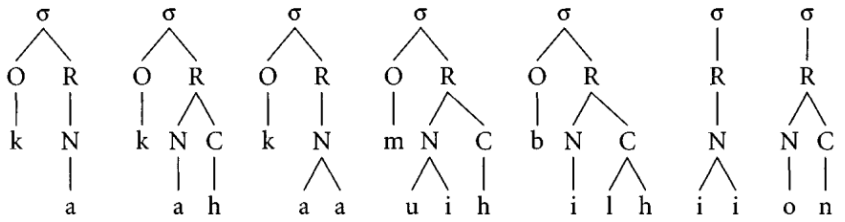
(1.19)

Word	*Possessive*	*Gloss*	*Word*	*Possessive*	*Gloss*
baa	baa.**ka**	'excrement'	bi.lam	bi.lam.**ka**	'fish'
dii.muih	dii.**ka**.muih	'snake'	gaad	gaad.**ka**	'god'
ii.bin	ii.**ka**.bin	'heaven'	ii.li.lih	ii.**ka**.li.lih	'shark'
ka h.ma	kah.**ka**.ma	'iguana'	ka.pak	ka.pak.**ka**	'manner'
lii.ma	lii.**ka**.ma	'lemon'	mis.tu	mis.**ka**.tu	'cat'
on.yan	on.**ka**.yan	'onion'	pau.mak	pau.**ka**.mak	'tomato'
sik.bilh	sik.**ka**.bilh	'horsefly'	taim	taim.**ka**	'time'
ta i.tai	tai.**ka**.tai	'grey squirrel'	uu.mak	uu.**ka**.mak	'window'
wa i.ku	wai.**ka**.ku	'moon, month'	wa.sa.la	wa.sa.**ka**.la	'possum'

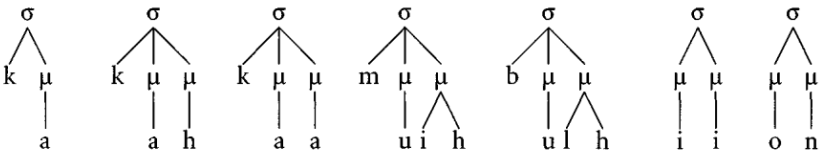
Observe that the infix appears at a syllable boundary, and so we can already state that the infix position is sensitive to syllable structure. Any analysis of the infix position must take **syllable weight** into consideration. Syllables having a single short vowel and no following consonants are defined to be **light**. (The presence of onset consonants is irrelevant to syllable weight.) All other syllables, i.e. those which have two vowels, or a single long vowel, or a final consonant, are defined to be **heavy**; e.g. *kah*, *kaa*, *muih*,

bilh, *ii*, *on*. Two common phonological representations for this syllable structure are the onset-rhyme model, and the moraic model. Representations for the syllables just listed are shown in (1.20). In these diagrams, σ denotes a syllable, O onset, R rhyme, N nucleus, C coda, and μ *mora* (the traditional, minimal unit of syllable weight).

(1.20) *a. The onset-rhyme model of syllable structure*



b. The moraic model of syllable structure



In the onset-rhyme model (1.20*a*), consonants coming before the first vowel are linked to the onset node, and the rest of the material comes under the rhyme node.[3] A rhyme contains an obligatory nucleus and an optional coda. In this model, a syllable is said to be heavy if and only if its rhyme or its nucleus are branching.

In the moraic mode (1.20*b*), any consonants that appear before the first vowel are linked directly to the syllable node. The first vowel is linked to its own mora node (symbolized by μ), and any remaining material is linked to the second mora node. A syllable is said to be heavy if and only if it has more than one mora.

These are just two of several ways that have been proposed for representing syllable structure. The syllables constituting a word can be linked to higher levels of structure, such as the *foot* and the *prosodic word*. For now, it is sufficient to know that such higher levels exist, and that we have a way to represent the binary distinction of syllable weight.

Now we can return to the Ulwa data, from example (1.19). A relatively standard way to account for the infix position is to stipulate that the first light syllable, if present, is actually invisible to the rules which assign syllables to higher levels; such syllables are said to be **extra-metrical**. They are a sort of 'upbeat' to the word, and are often associated with the preceding word in continuous speech. Given these general principles

[3] Two syllables usually have to agree on the material in their rhyme constituents in order for them to be considered rhyming, hence the name.

concerning hierarchical structure, we can simply state that the Ulwa possessive affix is infixed after the first syllable.[4]

In the foregoing discussion, I hope to have revealed many interesting issues which are confronted by phonological analysis, without delving too deeply into the abstract theoretical constructs which phonologists have proposed. Theories differ enormously in their organization of phonological information and the ways in which they permit this information to be subjected to rules and constraints, and the way the information is used in a lexicon and an overarching grammatical framework. Some of these theoretical frameworks include: lexical phonology, underspecification phonology, government phonology, declarative phonology, and optimality theory. For more information about these, please see section 1.5.3 for literature references.

1.5 Computational Phonology

When phonological information is treated as a string of atomic symbols, it is immediately amenable to processing using existing models. A particularly successful example is the work on finite-state transducers (see Chapter 18). However, phonologists abandoned linear representations in the 1970s, and so we will consider some computational models that have been proposed for multi-linear, hierarchical, phonological representations. It turns out that these pose some interesting challenges.

Early models of generative phonology, like that of the Sound Pattern of English (SPE), were sufficiently explicit that they could be implemented directly. A necessary first step in implementing many of the more recent theoretical models is to formalize them, and to discover the intended semantics of some subtle, graphical notations. A practical approach to this problem has been to try to express phonological information using existing, well-understood computational models. The principal models are finite-state devices and attribute-value matrices.

1.5.1 Finite-state models of non-linear phonology

Finite-state machines cannot process structured data, only strings, so special methods are required for these devices to process complex phonological representations. All approaches involve a many-to-one mapping from the parallel layers of representa-

[4] A better analysis of the Ulwa infixation data involves reference to **metrical feet**, phonological units above the level of the syllable. This is beyond the scope of the current chapter, however.

tion to a single machine. There are essentially three places where this many-to-one mapping can be situated. The first approach is to employ multi-tape machines (Kay 1987). Each tier is represented as a string, and the set of strings is processed simultaneously by a single machine. The second approach is to map the multiple layers into a single string, and to process that with a conventional single-tape machine (Kornai 1995). The third approach is to encode each layer itself as a finite-state machine, and to combine the machines using automaton intersection (Bird and Ellison 1994).

This work demonstrates how representations can be compiled into a form that can be directly manipulated by finite-state machines. Independently of this, we also need to provide a means for phonological generalizations (such as rules and constraints) to be given a finite-state interpretation. This problem is well studied for the linear case, and compilers exist that will take a rule formatted somewhat like the SPE style and produce an equivalent finite-state transducer. Whole constellations of ordered rules or optimality-theoretic constraints can also be compiled in this way. However, the compilation of rules and constraints involving autosegmental structures is still largely unaddressed.

The finite-state approaches emphasize the temporal (or left-to-right) ordering of phonological representations. In contrast, attribute-value models emphasize the hierarchical nature of phonological representations.

1.5.2 Attribute-value matrices

The success of attribute-value matrices (AVMs) as a convenient formal representation for constraint-based approaches to syntax (see Chapter 3), and concerns about the formal properties of non-linear phonological information, led some researchers to apply AVMs to phonology. Hierarchical structures can be represented using AVM nesting, as shown in (1.21*a*), and autosegmental diagrams can be encoded using AVM indices, as shown in (1.21*b*).

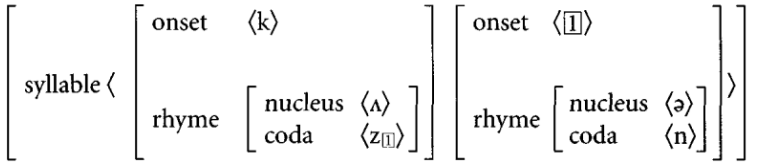
(1.21) *a.*
$$\begin{bmatrix} \text{onset} & \langle \text{k} \rangle \\ \text{rhyme} & \begin{bmatrix} \text{nucleus} & \langle \text{u, i} \rangle \\ \text{coda} & \langle \text{h} \rangle \end{bmatrix} \end{bmatrix}$$

b.
$$\begin{bmatrix} \text{syllable} & \langle \text{i}_{\boxed{1}}, \text{bu}f, \text{la}_{\boxed{3}}, \text{li}_{\boxed{4}} \rangle \\ \text{tone} & \langle \text{H}_{\boxed{5}}, \text{L}_{\boxed{6}}, \text{H}_{\boxed{7}}, \text{L}_{\boxed{8}} \rangle \\ \text{associations} & \{\langle \boxed{1}, \boxed{5} \rangle, \langle f, \boxed{5} \rangle, \langle \boxed{3}, \boxed{7} \rangle, \langle \boxed{4}, \boxed{8} \rangle\} \end{bmatrix}$$

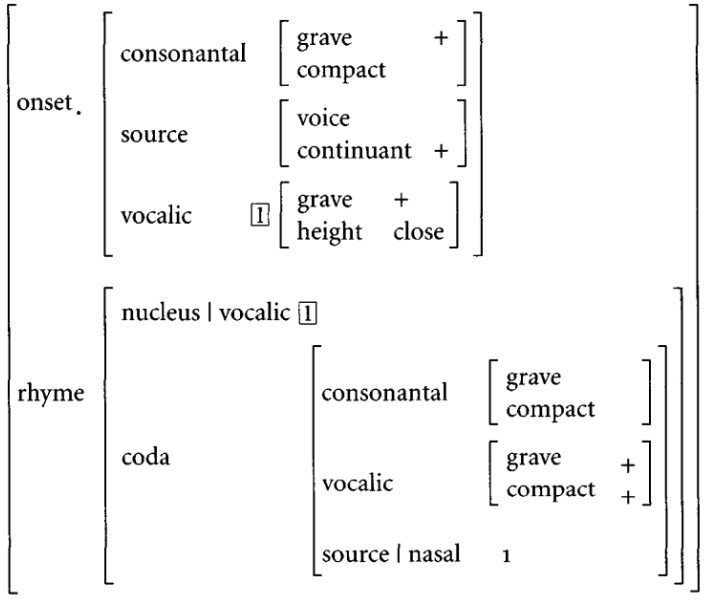
AVMs permit re-entrancy by virtue of the numbered indices, and so parts of a hierarchical structure can be shared. For example, (1.22*a*) illustrates a consonant shared

between two adjacent syllables, for the word *cousin* (this kind of double affiliation is called **ambisyllabicity**). Example (1.22*b*) illustrates shared structure within a single syllable *full*, to represent the **coarticulation** of the onset consonant with the vowel.

(1.22) *a.*



b.



Given such flexible and extensible representations, rules and constraints can manipulate and enrich the phonological information. Computational implementations of these AVM models have been used in speech synthesis systems.

1.5.3 Computational tools for phonological research

Once a phonological model is implemented, it ought to be possible to use the implementation to evaluate theories against data sets. A phonologist's workbench should help people to 'debug' their analyses and spot errors before going to press with an analysis. Developing such tools is much more difficult than it might appear.

First, there is no agreed method for modelling non-linear representations, and each proposal has shortcomings. Second, processing data sets presents its own set of problems, having to do with tokenization, symbols which are ambiguous as to their featural decomposition, symbols marked as uncertain or optional, and so on. Third, some innocuous-looking rules and constraints may be surprisingly difficult to model, and it might only be possible to approximate the desired behaviour. Additionally, certain universal principles and tendencies may be hard to express in a formal manner. A final, pervasive problem is that symbolic transcriptions may fail to adequately reflect linguistically significant acoustic differences in the speech signal.

Nevertheless, whether the phonologist is sorting data, or generating helpful tabulations, or gathering statistics, or searching for a (counter-)example, or verifying the transcriptions used in a manuscript, the principal challenge remains a computational one. Recently, new directed-graph models (e.g. Emu, MATE, Annotation Graphs) appear to provide good solutions to the first two problems, while new advances on finite-state models of phonology are addressing the third problem. Therefore, we have grounds for confidence that there will be significant advances on these problems in the near future.

Further Reading and Relevant Resources

The phonology community is served by an excellent journal *Phonology*, published by Cambridge University Press. Useful textbooks and collections include: Katamba (1989); Frost and Katz (1992); Kenstowicz (1994); Goldsmith (1995); Clark and Yallop (1995); Gussenhoven and Jacobs (1998); Goldsmith (1999); Roca, Johnson, and Roca (1999); Jurafsky and Martin (2000); Harrington and Cassidy (1999). Oxford University Press publishes a series *The Phonology of the World's Languages*, including monographs on Armenian (Vaux 1998), Dutch (Booij 1995), English (Hammond 1999), German (Wiese 1996), Hungarian (Siptár and Törkenczy 2000), Kimatuumbi (Odden 1996), Norwegian (Kristoffersen 1996), Portuguese (Mateus and Andrade 2000), and Slovak (Rubach 1993). An important survey of phonological variation is the *Atlas of North American English* (Labov et al. 2001).

Phonology is the oldest discipline in linguistics and has a rich history. Some historically important works include: Joos (1957); Pike (1947); Firth (1952); Bloch (1948); Hockett (1955); Chomsky and Halle (1968). The most comprehensive history of phonology is Anderson (1985).

Useful resources for phonetics include: Catford (1988); Laver (1994); Ladefoged and Maddieson (1996); Stevens (1999); International Phonetic Association (1999); Ladefoged (2000); Handke (2001), and the homepage of the International Phonetic Association http://www.arts.gla.ac.uk/IPA/ipa.html. The phonology/phonetics interface is an area of vigorous research, and the main focus of the *Laboratory Phonology*

series published by Cambridge University Press: Kingston and Beckman (1991); Docherty and Ladd (1992); Keating (1994); Connell and Arvaniti (1995); Broe and Pierrehumbert (2000). Two interesting essays on the relationship between phonetics and phonology are Pierrehumbert (1990); Fleming (2000).

Important works on the syllable, stress, intonation, and tone include the following: Pike and Pike (1947); Liberman and Prince (1977); Burzio (1994); Hayes (1994); Blevins (1995); Ladd (1996); Hirst and Di Cristo (1998); Hyman and Kisseberth (1998); van der Hulst and Ritter (1999). Studies of partial specification and redundancy include: Archangeli (1988); Broe (1993); Archangeli and Pulleyblank (1994).

Attribute-value and directed graph models for phonological representations and constraints are described in the following papers and monographs: Bird and Klein (1994); Bird (1995); Coleman (1998); Scobbie (1998); Bird and Liberman (2001); Cassidy and Harrington (2001).

The last decade has seen two major developments in phonology, both falling outside the scope of this limited chapter. On the theoretical side, Alan Prince, Paul Smolensky, John McCarthy, and many others have developed a model of constraint interaction called *Optimality Theory* (OT) (Archangeli and Langendoen 1997; Kager 1999; Tesar and Smolensky 2000). The Rutgers Optimality Archive houses an extensive collection of OT papers (http://ruccs.rutgers.edu/roa.html). On the computational side, the Association for Computational Linguistics (ACL) has a special interest group in computational phonology (SIGPHON) with a homepage at http://www.cogsci.ed.ac.uk/sigphon/. The organization has held five meetings to date, with proceedings published by the ACL and many papers available on-line from the SIGPHON site: Bird (1994*b*); Sproat (1996); Coleman (1997); Ellison (1998); Eisner et al. (2000). Another collection of papers was published as a special issue of the journal *Computational Linguistics* in 1994 (Bird 1994*a*). Several Ph.D. theses on computational phonology have appeared: Bird (1995); Kornai (1995); Tesar (1995); Carson-Berndsen (1997); Walther (1997); Boersma (1998); Wareham (1999); Kiraz (2000). Key contributions to computational OT include the proceedings of the fourth and fifth SIGPHON meetings and Ellison (1994); Tesar (1995); Eisner (1997); Karttunen (1998).

The sources of data published in this chapter are as follows: Russian (Kenstowicz and Kisseberth 1979); Chakosi (Ghana: Language Data Series, MS); Ulwa (Sproat 1992).

Acknowledgements

I am grateful to D. Robert Ladd and Eugene Buckley for comments on an earlier version of this chapter, and to James Roberts for furnishing me with the Chakosi data.

References

Anderson, S. R. 1985. *Phonology in the Twentieth Century: Theories of Rules and Theories of Representations*. Chicago: University of Chicago Press.

Archangeli, D. 1988. 'Aspects of underspecification theory'. *Phonology*, 5, 183–207.

——and Langendoen, D. T. (eds.). 1997. *Optimality Theory: An Overview*. Oxford: Blackwell.

——and D. Pulleyblank. 1994. *Grounded Phonology*. Cambridge, Mass.: MIT Press.

Bird, S. (ed.). 1994*a*. *Computational Linguistics: Special Issue on Computational Phonology*, 20(3).

——(ed.). 1994*b*. *Proceedings of the 1st Meeting of the Association for Computational Linguistics Special Interest Group in Computational Phonology (ACL '94)* (Las Cruces, N. Mex.).

——1995. *Computational Phonology: A Constraint-Based Approach*. Studies in Natural Language Processing. Cambridge: Cambridge University Press.

——1999. 'Dschang syllable structure'. In H. van der Hulst and N. Ritter (eds.), *The Syllable: Views and Facts*, Studies in Generative Grammar, Berlin: Mouton de Gruyter, 447–76.

——and T. M. Ellison. 1994. 'One level phonology: autosegmental representations and rules as finite automata'. *Computational Linguistics*, 20, 55–90.

——and E. Klein. 1994. 'Phonological analysis in typed feature systems'. *Computational Linguistics*, 20, 455–91.

——and M. Liberman. 2001. 'A formal framework for linguistic annotation'. *Speech Communication*, 33, 23–60.

Blevins, J. 1995. 'The syllable in phonological theory'. In J. A. Goldsmith (ed.), *The Handbook of Phonological Theory*. Cambridge, Mass.: Blackwell, 206–44.

Bloch, B. 1948. 'A set of postulates for phonemic analysis'. *Language*, 24, 3–46.

Boersma, P. 1998. *Functional phonology: formalizing the interactions between articulatory and perceptual drives*. Ph.D. thesis, University of Amsterdam.

Booij, G. 1995. *The Phonology of Dutch*. The Phonology of the World's Languages. Oxford: Clarendon Press.

Broe, M. 1993. 'Specification theory: the treatment of redundancy in generative phonology'. Ph.D. thesis, University of Edinburgh.

——and J. Pierrehumbert (eds.). 2000. *Papers in Laboratory Phonology*, v: *Language Acquisition and the Lexicon*. Cambridge: Cambridge University Press.

Burzio, L. 1994. *Principles of English Stress*. Cambridge: Cambridge University Press.

Carson-Berndsen, J. 1997. *Time Map Phonology: Finite State Models and Event Logics in Speech Recognition*. Text, Speech and Language Technology 5. Dordrecht: Kluwer.

Cassidy, S. and J. Harrington. 2001. 'Multi-level annotation of speech: an overview of the Emu speech database management system'. *Speech Communication*, 33, 61–77.

Catford, J. C. 1988. *Practical Introduction to Phonetics*. Oxford: Clarendon Press.

Chomsky, N. and M. Halle. 1968. *The Sound Pattern of English*. New York: Harper and Row.

Clark, J. and C. Yallop. 1995. *An Introduction to Phonetics and Phonology*. Oxford: Blackwell.

Coleman, J. (ed.). 1997. *Proceedings of the 3rd Meeting of the Association for Computational Linguistics Special Interest Group in Computational Phonology (ACL '97)* (Madrid).

——1998. *Phonological Representations: Their Names, Forms and Powers*. Cambridge Studies in Linguistics. Cambridge: Cambridge University Press.

Connell, B. and A. Arvaniti. 1995. *Papers in Laboratory Phonology*, iv: *Phonology and Phonetic Evidence*. Cambridge: Cambridge University Press.

Docherty, G. J. and D. R. Ladd (eds.). 1992. *Papers in Laboratory Phonology* ii: *Gesture, Segment, Prosody*. Cambridge: Cambridge University Press.

Eisner, J. 1997. 'Efficient generation in primitive optimality theory'. *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics* (*ACL '97*) (Madrid), 313–20.

——L. Karttunen, and A. Thériault (eds.). 2000. *Proceedings of the 5th Meeting of the Association for Computational Linguistics Special Interest Group in Computational Phonology (ACL 2000)* (Luxembourg).

Ellison, T. M. 1994. 'Phonological derivation in optimality theory'. *Proceedings of the 15th International Conference on Computational Linguistics* (*COLING '94*) (Kyoto), 1007–13.

——(ed.). 1998. *Proceedings of the 4th Meeting of the Association for Computational Linguistics Special Interest Group in Computational Phonology* (Quebec).

Firth, J. R. 1957. 'Sounds and prosodies'. In *Papers in Linguistics 1934–1951*. London: Clarendon Press, 121–38. First pub. 1948.

Fleming, E. 2000. 'Scalar and categorical phenomena in a unified model of phonetics and phonology'. *Phonology*, 1, 7–44.

Frost, R. and L. Katz (eds.). 1992. *Orthography, Phonology, Morphology and Meaning*. Advances in Psychology 94. Amsterdam: North-Holland.

Goldsmith, J. A. (ed.). 1995. *The Handbook of Phonological Theory*. Cambridge, Mass.: Blackwell.

——(ed.). 1999. *Phonological Theory: The Essential Readings*. Cambridge, Mass.: Blackwell.

Gussenhoven, C. and H. Jacobs. 1998. *Understanding Phonology*. London: Edward Arnold.

Hammond, M. 1999. *The Phonology of English: A Prosodic Optimality-Theoretic Approach*. Oxford: Clarendon Press.

Handke, J. 2001. *The Mouton Interactive Introduction to Phonetics and Phonology*. Berlin: Mouton de Gruyter.

Harrington, J. and S. Cassidy. 1999. *Techniques in Speech Acoustics*. Dordrecht: Kluwer.

Hayes, B. 1994. *Metrical Stress Theory: Principles and Case Studies*. Chicago: University of Chicago Press.

Hirst, D. and A. Di Cristo (eds.). 1998. *Intonation Systems: A Survey of Twenty Languages*. Cambridge: Cambridge University Press.

Hockett, C. F. 1955. *A Manual of Phonology*. Baltimore: Waverly Press.

Hyman, L. M. and C. Kisseberth (eds.). 1998. *Theoretical Aspects of Bantu Tone*. Stanford, Calif.: CSLI Publications.

International Phonetic Association 1999. *Handbook of the International Phonetic Association: A Guide to the Use of the International Phonetic Alphabet*. Cambridge: Cambridge University Press.

Joos, M. (ed.). 1957. *Readings in Linguistics*, i: *The Development of Descriptive Linguistics in America, 1925–56*. Chicago: University of Chicago Press.

Jurafsky, D. and J. H. Martin. 2000. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Prentice Hall.

Kager, R. 1999. *Optimality Theory*. Cambridge: Cambridge University Press.

Karttunen, L. 1998. *The proper treatment of optimality in computational phonology*. http://xxx.lanl.gov/abs/cmp-lg/9804002.

Katamba, F. 1989. *An Introduction to Phonology*. Reading, Mass: Addison Wesley.

Kay, M. 1987. 'Nonconcatenative finite-state morphology'. *Proceedings of the 3rd Meeting of the European Chapter of the Association for Computational Linguistics* (*ACL '87*) (Copenhagen), 2–10.

Keating, P. A. 1994. *Papers in Laboratory Phonology*, iii: *Phonological Structure and Phonetic Form*. Cambridge: Cambridge University Press.

Kenstowicz, M. 1994. *Phonology in Generative Grammar*. Oxford: Blackwell.

——and C. Kisseberth. 1979. *Generative Phonology: Description and Theory*. New York: Academic Press.

Kingston, J. and M. E. Beckman (eds.). 1991. *Papers in Laboratory Phonology*, i: *Between the Grammar and the Physics of Speech*. Cambridge: Cambridge University Press.

Kiraz, G. 2000. *Computational Approach to Non-linear Morphology*. Studies in Natural Language Processing. Cambridge: Cambridge University Press.

Kornai, A. 1995. *Formal Phonology*. New York: Garland Publishing.

Kristoffersen, G. 1996. *The Phonology of Norwegian*. The Phonology of the World's Languages. Oxford: Clarendon Press.

Labov, W., S. Ash, and C. Boberg. 2001. *Atlas of North American English*. Berlin: Mouton de Gruyter.

Ladd, D. R. 1996. *Intonational Phonology*. Cambridge: Cambridge University Press.

Ladefoged, P. 2000. *Vowels and Consonants: An Introduction to the Sounds of Languages*. Cambridge, Mass.: Blackwell.

——and I. Maddieson. 1996. *The Sounds of the World's Languages*. Cambridge, Mass.: Blackwell.

Laver, J. 1994. *Principles of Phonetics*. Cambridge Textbooks in Linguistics. Cambridge: Cambridge University Press.

Liberman, M. Y. and A. S. Prince. 1977. 'On stress and linguistic rhythm'. *Linguistic Inquiry*, 8, 249–336.

Mateus, H. and E. d'Andrade. 2000. *The Phonology of Portuguese*. The Phonology of the World's Languages. Oxford: Clarendon Press.

Odden, D. 1996. *The Phonology and Morphology of Kimatuumbi*. The Phonology of the World's Languages. Oxford: Clarendon Press.

Pierrehumbert, J. 1990. 'Phonological and phonetic representation'. *Journal of Phonetics*, 18, 375–94.

Pike, K. L. 1947. *Phonemics: A Technique for Reducing Language to Writing*. Ann Arbor: University of Michigan Press.

——and E. V. Pike. 1947. 'Immediate constituents of Mazateco syllables'. *International Journal of American Linguistics*, 13, 78–91.

Roca, I., W. Johnson, and A. Roca. 1999. *A Course in Phonology*. Cambridge, Mass.: Blackwell.

Rubach, J. 1993. *The Lexical Phonology of Slovak*. Oxford: Clarendon Press.

Scobbie, J. 1998. *Attribute-Value Phonology*. New York: Garland Publishing.

Siptár, P. and M. Törkenczy. 2000. *The Phonology of Hungarian*. The Phonology of the World's Languages. Oxford: Clarendon Press.

Sproat, R. 1992. *Morphology and Computation*. Natural Language Processing. Cambridge, Mass.: MIT Press.

——(ed.). 1996. *Computational Phonology in Speech Technology: Proceedings of the 2nd Meeting of the Association for Computational Linguistics Special Interest Group in Computational Phonology (ACL '96)*. (Santa Cruz).

Stevens, K. N. 1999. *Acoustic Phonetics*. Cambridge, Mass.: MIT Press.

Tesar, B. 1995. *Computational optimality theory*. Ph.D. thesis, Rutgers University.

——and P. Smolensky. 2000. *Learnability in Optimality Theory*. Cambridge, Mass.: MIT Press.

van der Hulst, H. and N. Ritter (eds.). 1999. *The Syllable: Views and Facts*. Studies in Generative Grammar. Berlin: Mouton de Gruyter.

Vaux, B. 1998. *The Phonology of Armenian*. The Phonology of the World's Languages. Oxford: Clarendon Press.

Walther, M. 1997. *Declarative prosodic morphology: constraint-based analyses and computational models of Finnish and Tigrinya*. Ph.D. thesis, Heinrich-Heine-Universität, Düsseldorf (in German).

Wareham, T. 1999. *Systematic parameterized complexity analysis in computational phonology*. Ph.D. thesis, University of Victoria.

Wiese, R. 1996. *The Phonology of German*. The Phonology of the World's Languages. Oxford: Clarendon Press.

CHAPTER 2

MORPHOLOGY

HARALD TROST

ABSTRACT

Computational morphology deals with the processing of words in both their graphemic, i.e. written, and their phonemic, i.e. spoken form. It has a wide range of practical applications. Probably every one of you has already come across some of them. Ever used spelling correction? Or automated hyphenation? This is computational morphology at work. These tasks may seem simple to a human but they pose hard problems to a computer program. This chapter will provide you with insights into why this is so and what techniques are available to tackle these tasks.

2.1 LINGUISTIC BACKGROUND

Natural languages have intricate systems to create words and word forms from smaller units in a systematic way. The part of linguistics concerned with these phenomena is morphology. This chapter starts with a quick overview of this fascinating field. The account given will be mostly pre-theoretic and purely descriptive. Readers interested in morphological theory should consult the Further Reading section.

What is morphology all about? A simple answer is that morphology deals with words. In formal language words are just arbitrary strings denoting constants or variables. Nobody cares about a morphology of formal languages. In contrast, human

languages contain some hundreds of thousands of words, each describing some particular feature of our world. Continuously new words are integrated while others are drifting out of use. This infinity of words is produced from a finite collection of smaller units. The task of morphology is to find and describe the mechanisms behind this process.

The basic building blocks are **morphemes**, defined as the smallest unit in language to which a meaning may be assigned or, alternatively, as the minimal unit of grammatical analysis. Morphemes are abstract entities expressing basic features, either semantic concepts like *door*, *blue*, or *take* which are called **roots** or abstract features like *past* or *plural*.

Their realization as part of a word is called **morph**. Often, there is a one-to-one relation, e.g. the morpheme *door* is realized as the morph *door*. With *take*, on the other hand, we find the morphs *take* and *took*. In such a case we speak of **allomorphs**. Plural in English is usually expressed by the morph *-s*. There are exceptions though: in *oxen* plural is expressed through the morph *-en*, in *men* by stem vowel alteration. All these different forms are allomorphs of the plural morpheme.

Free morphs may form a word on their own, e.g. the morph *door*. Such words are **monomorphemic**, i.e. they consist of a single morph. **Bound** morphs occur only in combination with other forms. All affixes are bound morphs. For example, the word *doors* consists of the free morph *door* and the bound morph *-s*. Words may also consist of free morphs only, e.g. *tearoom*, or bound morphs only, e.g. *exclude*.

Every language typically contains some 10,000 morphs. This is a magnitude below the number of words. Strict rules govern the combination of these morphs to form words (cf. section 2.5). This way of structuring the lexicon makes the cognitive load of remembering so many words much easier.

2.2 What is a Word?

Surprisingly, there is no straight answer to this question. One can easily spot 'words' in a text because they are separated from each other by blanks or punctuation. However, if you record ordinary speech you will find out that there are no obvious breaks. On the other hand, we could isolate units occurring—in different combinations—over and over again. Therefore, the notion of 'word' makes sense. How can we define it?

From a syntactic point of view, 'words' are the units that make up sentences. Words are grouped according to their function in the sentential structure. Morphology, on the other hand, is concerned with the inner structure of 'words'. It tries to uncover the rules that govern the formation of words from smaller units. We notice that words

that convey the same meaning look different depending on their syntactic context. Take, e.g., the words *degrade*, *degrades*, *degrading*, and *degraded*. We can think of those as different forms of the same 'word'. The part that carries the meaning is the **base form**. In our example this is the form *degrade*. All other forms are produced by combination with additional morphs. All the different forms of a word together are called its **paradigm**.

In English, the base form always coincides with a specific word form, e.g. degrade is also present tense, active voice, non-third person singular. In other languages we find a slightly different situation. Italian marks nouns for gender and number. Different affixes are used to signal masculine and feminine on the one hand and singular and plural on the other hand.

(2.1)

	Singular	*Plural*	
Masculine	pomodoro	pomodori	'tomato'
Feminine	cipolla	cipolle	'onion'

We must assume that the base form is what is left over after removing the respective suffixes, i.e. *pomodor-* and *cipoll-*. Such base forms are called **stems**.

Base forms are not necessarily atomic. By comparing *degrade* to *downgrade*, *retrograde*, and *upgrade* on the one hand and *decompose*, *decrease*, and *deport* on the other hand, we notice that *degrade* is composed of the morphs *de-* and *grade*. The morpheme carrying the central meaning of the word is often called the **root**. Roots may combine with affixes or other roots (cf. section 2.3.2) to form new base forms.

In phonology 'words' define the range for certain phonological processes. Often the phonological word is identical with the morphological word but sometimes boundaries differ. A good example for such a discrepancy is cliticization (cf. section 2.5.2).

2.3 Functions of Morphology

How much and what sort of information is expressed by morphology differs widely between languages. Information that is expressed by syntax in one language is expressed morphologically in another one. For example, *English* uses an auxiliary verb construction, *Spanish* a suffix to express the future tense.

(2.2) I speak—hablo
I will speak—hablaré

Also, some type of information may be present in one language while missing in another one. For example, many languages mark nouns for plural. Japanese does not.

(2.3) book—hon
books—hon

The means for encoding information vary widely. Most common is the use of different types of affixes. Traditionally, linguists discriminate between the following types of languages:

- **Isolating languages** (e.g. Mandarin Chinese): there are no bound forms, e.g. no affixes. The only morphological operation is composition.
- **Agglutinative languages** (e.g. Ugro-Finnic and Turkic languages): all bound forms are affixes, i.e. are added to a stem like beads on a string. Every affix represents a distinct morphological feature. Every feature is expressed by exactly one affix.
- **Inflectional languages** (e.g. Indo-European languages): distinct features are merged into a single bound form (portmanteau morph). The same underlying feature may be expressed differently, depending on the paradigm.
- **Polysynthetic languages** (e.g. Inuit languages): these languages express more structural information morphologically than other languages, e.g. verb arguments are incorporated into the verb.

Real languages rarely fall cleanly into one of the above classes, e.g. even Mandarin has a few suffixes. Moreover, this classification mixes the aspect of what is expressed morphologically and the means for expressing it.

2.3.1 Inflection

Inflection is required in particular syntactic contexts. It does not change the part-of-speech category but the grammatical function. The different forms of a word produced by inflection form its **paradigm**. Inflection is *complete*, i.e. with rare exceptions all the forms of its paradigm exist for a specific word. Regarding inflection, words can be categorized in three classes:

- **Particles** or non-inflecting words: they occur in just one form. In English, prepositions, adverbs, conjunctions, and articles are particles;
- **Verbs** or words following conjugation;
- **Nominals** or words following declination, i.e. nouns, adjectives, and pronouns.

Conjugation is mainly concerned with defining tense, aspect, and agreement (e.g. person and number). Take for example the *German* verb 'lesen' (to read):

(2.4)

	Present				*Past*			
	Indicative		*Subjunctive*		*Indicative*		*Subjunctive*	
	Sing.	*Plural*	*Sing.*	*Plural*	*Sing.*	*Plural*	*Sing.*	*Plural*
1st person	lese	lesen	lese	lesen	las	lasen	läse	läsen
2nd person	liest	lest	lesest	leset	last	last	läsest	läset
3rd person	liest	lesen	lese	lesen	las	lasen	läse	läsen
Participle	lesend				gelesen			
Imperative	lies	lest						
Infinitive	lesen							

Declination marks various agreement features like *number* (singular, plural, dual, etc.), *case* (as governed by verbs and prepositions, or to mark various kinds of semantic relations), *gender* (male, female, neuter), and *comparison*.

2.3.2 Derivation and compounding

Derivation and compounding are processes that create *new words*. They have nothing to do with morphosyntax but are a means to extend our lexicon in an economic and principled way.

In **derivation**, a new word—usually of a different part-of-speech category—is produced by adding a bound morph to a base form. Derivation is incomplete, i.e. a derivational morph cannot be applied to all words of the appropriate class. For example, in German the very productive derivational suffix *-bar* can be applied to most but not all verbs to produce adjectives:

(2.5)

essen	'eat'	—	essbar	'eatable'
absehen	'conceive'	—	absehbar	'conceivable'
sehen	'see'	—	*sehbar	'visible'

Application of a derivational morph may be restricted to a certain subclass. For example, the English derivational suffix *-ity* combines with stems of Latin origin only, while the Germanic suffix *-ness* applies to a wider range:

(2.6)

rare	—	rarity	—	rareness
red	—	*reddity	—	redness
grave	—	gravity	—	graveness
weird	—	*weirdity	—	weirdness

Derivation can be applied recursively, i.e. words that are already the product of derivation can undergo the process again. That way a potentially infinite number of words can be produced. Take, for example, the following chain of derivations:

(2.7) hospital—hospitalize—hospitalization—pseudohospitalization

Semantic interpretation of the derived word is often difficult. While a derivational

suffix can usually be given a unique semantic meaning, many of the derived words may still resist compositional interpretation.

While inflectional and derivational morphology are mediated by the attachment of a bound morph, **compounding** is the joining of two or more base forms to form a new word as in *state monopoly*, *bedtime*, or *red wine*. In some cases parts are joined by a linking morph (usually the remnant of case marking) as in *bull's eye* or German *Liebeslied* (love-song).

The last part of a compound usually defines its morphosyntactic properties. Semantic interpretation is even more difficult than with derivation. Almost any semantic relationship may hold between the components of a compound:

(2.8) Wienerschnitzel 'cutlet Vienna style'
Schweineschnitzel 'pork cutlet'
Kinderschnitzel 'cutlet for children'

The boundary between derivation and compounding is fuzzy. Historically, most derivational suffixes developed from words frequently used in compounding. An obvious example is the *-ful* suffix as in *hopeful*, *wishful*, *thankful*.

Phrases and compounds cannot always be distinguished. The English expression *red wine* in its written form could be both. In spoken language the stress pattern differs: *red wíne* vs. *réd wine*. In German phrases are morphologically marked, while compounds are not: *roter Wein* vs. *Rotwein*. For verb compounds the situation is similar to English: *zu Hause bleiben* vs. *zuhausebleiben*.

2.4 What Constitutes a Morph?

Every word form must at the core contain a root which can (must) then be complemented with additional morphs. How are these morphs realized? Obviously, a morph must somehow be recognizable in the phonetic or orthographic pattern constituting the word. The most common type of morph is a continuous sequence of phonemes. All roots and most affixes are of this form. A complex word then consists of a sequence of concatenated morphs. Agglutinative languages function almost exclusively this way. But there are surprisingly many other possibilities.

2.4.1 Affixation

An **affix** is a bound morph that is realized as a sequence of phonemes (or graphemes).

By far the most common types of affixes are prefixes and suffixes. Many languages have only these two types of affixes. Among them is English (at least under standard morphological analyses).

A **prefix** is an affix that is attached in front of a stem. An example is the English negative marker *un-* attached to adjectives:

(2.9) common uncommon

A **suffix** is an affix that is attached after a stem, e.g. the English plural marker *-s*:

(2.10) shoe shoes

Across languages suffixation is far more frequent than prefixation. Also, certain kinds of morphological information are never expressed via prefixes, e.g. nominal case marking. Many computational systems for morphological analysis and generation assume a model of morphology based on prefixation and suffixation only.

A **circumfix** is the combination of a prefix and a suffix which together express some feature. From a computational point of view a circumfix can be viewed as really two affixes applied one after the other.

In *German*, the circumfixes *ge—t* and *ge—n* form the past participle of verbs:

(2.11) sag*en* 'to say' gesagt 'said'
laufen 'to run' gelaufen 'run'

An **infix** is an affix where the placement is defined in terms of some phonological condition(s). These might result in the infix appearing within the root to which it is affixed. In *Bontoc* (Philippines) the infix *-um-* turns adjectives and nouns into verbs (Fromkin and Rodman 1997: 129). The infix attaches after the initial consonant:

(2.12) /fikas/ 'strong' /fumikas/ 'to be strong'
/kilad/ 'red' /kumilad/ 'to be red'
/fusul/ 'enemy' /fumusul/ 'to be an enemy'

Reduplication is a border case of affixation. The form of the affix is a function of the stem to which it is attached, i.e. it copies (some portion of) the stem. Reduplication may be complete or partial. In the latter case it may be prefixal, infixal, or suffixal. Reduplication can include phonological alteration on the copy or the original.

In *Javanese* **complete reduplication** expresses the *habitual-repetitive*. If the second vowel is non-/a/, the first vowel in the copy is made non-low and the second becomes /a/. When the second vowel is /a/, the copy remains unchanged while in the original the /a/ is changed to /ɛ/ (Kiparsky 1987):

(2.13) /adʊs/ 'take a bath' /odasadʊs/
/bali/ 'return' /bolabali/
/bozən/ 'tired of' /bozanbozən/
/ɛlɛq/ 'return' /elaqɛlɛq/
/dolan/ 'recreate' /dolandolɛn/
/udan/ 'horse' /udanudɛn/

Partial reduplication is more common. In *Yidiŋ* (Australia) **prefixal reduplication** by copying the 'minimal word' is used for plural marking (Nash 1980).

(2.14)	/mulari/	'initiated man'	/mulamulari/
	/gindalba/	'lizard'	/gindalgindalba/

In *Amharic* (Ethiopia) **infixal reduplication** is used to express the *frequentative* (Rose 2001).

(2.15)	/kətəfə/	'chop'	/kitatəfə/	'chop a lot'
	/k'əbələ/	'decrease'	/k'ibabələ/	'decrease greatly'
	/wək'ət'ə/	'fight'	/wik'ak'ət'ə/	'fight a lot'
	/lak'ət'ə/	'mix'	/lik'ak'ət'ə/	'mix a lot'

From a computational point of view one property of reduplication is especially important: since reduplication involves copying it cannot—at least in the general case—completely be described with the use of finite-state methods.

2.4.2 Non-concatenative phenomena

Semitic languages (at least according to standard analyses) exhibit a very peculiar type of morphology, often called **root-and-template morphology**. A so-called root, consisting of two to four consonants, conveys the basic semantic meaning. A vowel pattern marks information about voice and aspect. A derivational template gives the class of the word. *Arabic* verb stems are constructed this way. The root *ktb* (write) produces—among others—the following stems:

(2.16)	*Template*	*Vowel pattern*		
		A (active)	*UI (passive)*	
	CVCVC	katab	kutib	'write'
	CVCCVC	kattab	kuttib	'cause to write'
	CVVCVC	ka:tab	ku:tib	'correspond'
	tVCVVCVC	taka:tab	tuku:tib	'write each other'
	nCVVCVC	nka:tab	nku:tib	'subscribe'
	CtVCVC	ktatab	ktutib	'write'
	stVCCVC	staktab	stuktib	'dictate'

Sometimes, morphs neither introduce new nor remove existing segments. Instead, they are realized as a change of phonetic properties or an alteration of prosodic shape.

Ablaut refers to vowel alternations inherited from Indo-European. It is a pure example of vowel modification as a morphological process. Examples are strong verbs in Germanic languages (e.g. swim—swam—swum). In *Icelandic* this process is still more common and more regular than in most other Germanic languages (Sproat 1992: 62):

(2.17)

Stem	*Past sing.*	*Past pl.*	*PPP*	
/bi:t/	/beit/	/bit/	/bit/	'to bite'
/ri:f/	/reif/	/rif/	/rif/	'to tear'

Umlaut has its origin in a phonological process, whereby root vowels were assimilated to a high-front suffix vowel. When this suffix vowel was lost later on, the change in the root vowel became the sole remaining mark of the morphological feature originally signalled by the suffix. In *German* noun plural may be marked by umlaut (sometimes in combination with a suffix), i.e. the stem vowel feature *back* is changed to *front*:

(2.18)

Singular	*Plural*	
Mutter /mʊtɐ/	Mütter /mʏtɐ/	'mother'
Garten /gartən/	Gärten /gɛrtən/	'garden'
Hof /ho:f/	Höfe /hø:fə/	'yard'

Altering the prosody can also realize a morpheme. **Tone modification** can signal certain morphological features. In *Ngbaka* (Congo) tense-aspect contrasts are expressed by four different tonal variants (Nida 1949):

(2.19)

Low	*Mid*	*Low-high*	*High*	
/à/	/ā/	/ǎ/	/á/	'put more than one thing'
/kpòlò/	/kpōlō/	/kpòló/	/kpóló/	'return'
/b'ìlì/	/b'īlī/	/b'ìlí/	/b'ílí/	'cut'

A morpheme may be realized by a **stress shift**. *English* noun–verb derivation sometimes uses a pattern where stress is shifted from the first to the second syllable:

(2.20)

Noun	*Verb*
éxport	expórt
récord	recórd
cónvict	convíct

Suppletion is a process of total modification occurring sporadically and idiosyncratically within inflectional paradigms. It is usually associated with forms that are used very frequently, for example *went*, the past tense of *to go*, and the forms of *to be*: *am*, *are*, *is*, *was*, and *were*.

Sometimes a morphological operation has no phonological expression whatsoever. Examples are found in many languages. *English* noun-to-verb derivation is often not explicitly marked:

(2.21)

man	The man smiled.	Man the boats.
house	He buys a house.	They house in a cave.

A possible analysis is to assume a **zero morph** which attaches to the noun to form a verb: book+$\emptyset_V$. Another possibility is to assume two independent lexical items disregarding any morphological relationship.

2.5 The Structure of Words: Morphotactics

Somehow morphs must be put together to form words. A word grammar determines the way this has to be done. This part of morphology is called **morphotactics**. As we have seen, the most usual way is simple concatenation. Let's have a look at the constraints involved. What are the conditions governing the ordering of morphemes in *pseudohospitalization*?

(2.22) *hospitalationizepseudo, *pseudoizehospitalation
(2.23) *pseudohospitalationize

In (2.22) an obvious restriction is violated: *pseudo-* is a prefix and must appear ahead of the stem, *-ize* and *-ation* are suffixes and must appear after the stem. The violation in (2.23) is less obvious. In addition to the pure ordering requirements there are also rules governing to which types of stems an affix may attach: *-ize* attaches to nouns and produces verbs, *-ation* attaches to verbs and produces nouns.

One possibility for describing the word-formation process is to assume a functor-argument structure. Affixes are functors that pose restrictions on their (single) argument. That way a binary tree is constructed. Prefixes induce right branching and suffixes left branching.

The functor *pseudo-* takes a noun to form a noun, *-ize* a noun to form a verb, and *-ation* a verb to form a noun. This description renders two different possible structures for *pseudohospitalization*, the one given in Fig. 2.1 and a second one where *pseudo-* combines directly with *hospital* first. We may or may not accept this ambiguity. To avoid the second reading we could state a lexical constraint that a word with the head *pseudo-* cannot serve as an argument anymore.

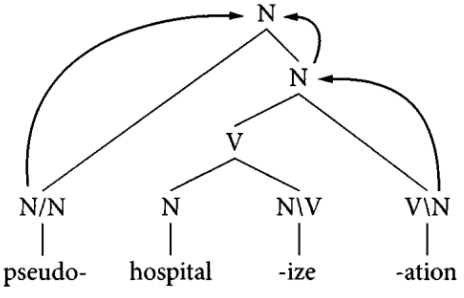


Fig. 2.1 The internal structure of the word *pseudohospitalization*

2.5.1 Constraints on affixes

Affixes attach to specific categories only. This is an example for a syntactic restriction. Restrictions may also be of a phonological, semantic, or purely lexical nature. A semantic restriction on the English adjectival prefix *un-* prevents its attachment to an adjective that already has a negative meaning:

(2.24)

unhappy	*unsad
unhealthy	*unill
unclean	*undirty

The fact that in English some suffixes may only attach to words of Latin origin (cf. section 2.3.2) is an example for a lexical restriction.

2.5.2 Morphological vs. phonological structure

In some cases there is a mismatch between the phonological and the morphological structure of a word. One example is comparative formation with the suffix *-er* in English. Roughly, there is a phonological rule that prevents attaching this suffix to words that consist of more than two syllables:

(2.25)

tall	taller
happy	happier
competent	*competenter
elegant	*eleganter

If we want to stick to the above rule *unrulier* has to be explained with a structure where the prefix *un-* is attached to *rulier*. But, from a morphological point of view, the adjective *ruly* does not exist, only the negative form *unruly*. This implies that the suffix *-er* is attached to *unruly*. We end up with an obvious mismatch!

A **clitic** is a syntactically separate word phonologically realized as an affix. The phenomenon is quite common across languages.

- English auxiliaries have contracted forms that function as affixes:
  he shall return → *he'll return*
- German prepositions can combine with the definite article:
  an dem Tisch → *am Tisch*
  in das Haus → *ins Haus*
- Italian personal pronouns can be attached to the verb. In this process the ordering of constituents is also altered:
  ce ne facciamo → *facciamocene*

2.6 The Influence of Phonology

Morphotactics is responsible for governing the rules for the combination of morphs into larger entities. One could assume that this is all a system needs to know to break down words into their component morphemes. But there is another aspect that makes things more complicated: Phonological rules may apply and change the shape of morphs. To deal with these changes and their underlying reasons is the area of **morphophonology**.

Most applications of computational morphology deal with text rather than speech. But written language is rarely a true phonemic description. For some languages, e.g. Finnish, Spanish, or Turkish, orthography is a good approximation for a phonetic transcription. English, on the other hand, has very poor correspondence between writing and pronunciation. As a result, we often deal with orthography rather than phonology. A good example is English plural rules (cf. section 2.8.1).

By and large, words are composed by concatenating morphs. Sometimes this concatenation process will induce a phonological change in the vicinity of the morph boundary.

Assimilation is a process where the two segments at a morph boundary influence each other, resulting in some feature change that makes them more similar. Take, for example, the English prefix *in-* where the *n* changes to *m* before labials:

(2.26) <in+feasible> → infeasible
<in+mature> → immature
<in+probable> → improbable
<in+secure> → insecure

Other possibilities are **epenthesis** (insertion) and **elision** (deletion) of a segment under certain (phonological) conditions. Take for example English plural formation:

(2.27) <door+s> → doors
<dish+s> → dishes
<bliss+s> → blisses
<match+s> → matches

In this case the rule requires the insertion of an /ə/ between /s/, /z/, /S/, or /Z/ and another /s/. On the other hand, the *German* suffix *-st* loses its starting segment /s/ when attached to stems ending in /s/:

(2.28) <leb+st> → lebst
<sag+st> → sagst
<ras+st> → rast
<trotz+st> → trotzt

The change is not purely phonologically motivated. The same condition, namely two

adjoining /s/, leads to either the epenthesis of /ə/, or the elision of the second /s/.[1]

Some morphophonological processes work long-distance. Most common are harmony rules. **Vowel harmony** is a phonological process where the leftmost (in rare cases the rightmost) vowel in a word influences all the following (preceding) vowels. It occurs in Finno-Ugric, Turkic, and many African languages. An example of Turkish vowel harmony is presented in section 1.2.

2.7 Applications of Computational Morphology

For **hyphenation** segmenting words correctly into their morphs is a prerequisite. The major problem is spurious segmentations. **Grapheme-to-phoneme conversion** for text to speech needs to resolve ambiguities in the translation of characters into phonemes. In the word *hothouse* we need to know the morph structure <hot+house> for correctly pronouncing the *th* sequence as /th/ instead of the usual /θ/ or /ð/.

Spelling correction is another low-level application. Comparing input against a list of word forms does not work well. The list will never contain all occurring words and enlarging the list has the negative side effect of including obscure words that will match with typos thus preventing their detection. Most current systems use a root lexicon, plus a relatively small set of affixes and simple rules to cover morphotactics.

Stemmers are used in information retrieval (see Chapter 29) to reduce as many related words and word forms as possible to a common canonical form—not necessarily the base form—which can then be used in the retrieval process. The main requirement is—as in all the above tasks—robustness.

In Chinese, Japanese, or Korean, words in a sentence are not separated by blanks or punctuation marks. Morphological analysis is used to perform the task of automatic word separation.

Written Japanese is a combination of kanji, the morphemic Chinese characters used for open-class morphemes, and the syllabic kana characters mainly used for closed-class morphemes (although in principle all Japanese words can be written exclusively in kana). Since there are several thousand kanji characters, many Japanese text input systems use kana–kanji conversion. The whole text is typed in kana and the relevant portions are subsequently converted to kanji. This mapping from kana

[1] The notion of insertion or deletion is purely descriptive. Phonological theory may explain the underlying processes completely differently. Nonetheless, this is the view most often taken by work in computational morphology.

to kanji is quite ambiguous. A combination of statistical and morphological methods is applied to solve the task.

An obvious application of computational morphology can be seen in systems based on parsing and/or generating utterances in written or spoken form. These range from message and information extraction to dialogue systems and machine translation. For many current applications, only inflectional morphology is considered.

In a parser, morphological analysis of words is an important prerequisite for syntactic analysis. Properties of a word the parser needs to know are its part-of-speech category and the morphosyntactic information encoded in the particular word form. Another important task is **lemmatization**, i.e. finding the corresponding dictionary form for a given input word, because for many applications a lemma lexicon is used to provide more detailed syntactic (e.g, valency) and semantic information for deep analysis. In generation, on the other hand, the task is to produce the correct word form from the base form plus the relevant set of morphosyntactic features.

At the moment most available speech recognition systems make use of full-form lexicons and perform their analysis on a word basis. Increasing demands on the lexicon size on the one hand and the need to limit the necessary training time on the other hand will make morph-based recognition systems more attractive.

2.8 Computational Morphology

The most basic task in computational morphology is to take a string of characters or phonemes as input and deliver an analysis as output. The input string (2.29) can be mapped to the string of underlying morphemes (2.30) or the morphosyntactic interpretation (2.31).

(2.29) incompatibilities
(2.30) in+con+patible+ity+s
(2.31) incompatibility+NounPlural

The simplest way to achieve the mapping from (2.29) to (2.31) is the **full-form lexicon**, i.e. a long list of pairs where each left side represents a word form and the right side its interpretation. Advantages are simplicity and applicability to all possible phenomena, disadvantages are redundancy and inability to cope with forms not contained in the lexicon.

Lemma lexicons reduce redundancy. A **lemma** is a canonical form—usually the base form—taken as the representative for all the different forms of a paradigm.[2] An

[2] This is also the approach taken in printed dictionaries.

interpretation algorithm relates every form to its lemma plus delivering a morphosyntactic interpretation. As a default, forms are expected to be string concatenations of lemma and affixes. Affixes must be stored in a separate repository together with the relevant morphotactic information about how they may combine with other forms. Interpretation simply means finding a sequence of affixes and a base form that conforms to morphotactics. For different reasons a given word form may not conform to this simple picture:

- With very frequently used words we find suppletion.

One needs some exception-handling mechanism to cope with suppletion. A possible solution is to have secondary entries where you store suppleted forms together with their morphosyntactic information. These secondary forms are then linked to the corresponding primary form, i.e. the lemma.

- Morphs are realized in a non-concatenative way, e.g. tense of strong verbs in English: *give–gave–given*.

In languages like English, where these phenomena affect only a fairly small and closed set of words, such forms can be treated like suppletion. Alternatively, some exception-handling mechanism (usually developed ad hoc and language specific) is applied.

- Phonological rules may change the shape of a word form, e.g. English suffixes starting with *s* may not directly follow stems ending in a sibilant: *dish–dishes*.

If morphophonological processes in a language are few and local the lemma lexicon approach can still be successful. In our example it suffices to assume two plural endings: *-s* and *-es*. For all base forms it must be specified whether the former or the latter of the two endings may be attached.

Apart from the obvious limitations with regard to the treatment of morphophonological rules on a more general scale the approach has some other inherent restrictions:

- The algorithm is geared towards analysis. For generation purposes, one needs a completely different algorithm and data.
- Interpretation algorithms are language specific because they encode both the basic concatenation algorithm and the specific exception-handling mechanism.
- The approach was developed for morphosyntactic analysis. An extension to handle more generally the segmenting of word forms into morphs is difficult to achieve.

2.8.1 Finite-state morphology

Because most morphological phenomena can be described with regular expressions the use of finite-state techniques for morphological components is common. In

particular, when morphotactics is seen as a simple concatenation of morphs it can straightforwardly be described by finite automata.

However, it was not so obvious how to describe non-concatenative phenomena (e.g. vowel harmony, root-and-template morphology, reduplication) and morphophonology in such a framework.

2.8.1.1 *Two-level morphology*

Two-level morphology explicitly takes care of morphophonology. The mechanism derives from the ideas developed in generative phonology (cf. Chomsky and Halle 1968). There, the derivation of a word form from its lexical structure is performed by the successive application of phonological rules creating a multi-step process involving several intermediate levels of representation. Such an approach may be suited for generation but leads to problems if applied to analysis. Since the ordering of rule application influences the result it is difficult to reverse the process.

Several proposals were made on how to overcome these problems. Two-level morphology (Koskenniemi 1984) is the most successful attempt. It has the further advantages of being non-directional (applicable to analysis and generation) and language independent (because of its purely declarative specification of language-specific data). Two-level morphology has since been implemented in a number of different systems and applied to a wide range of natural languages.

2.8.1.1.1 *Two-level rules*

As the name suggests, two levels suffice to describe the phonology (or orthography) of a natural language. On the surface level words appear just as they are pronounced (or written) in ordinary language (with the important exception of the null character). On the lexical level, the alphabet includes special symbols—so-called **diacritics**—which are mainly used to represent features that are not phonemes (or graphemes) but nevertheless constitute necessary phonological information. The diacritics '+' and '#' are used to indicate morph and word boundary respectively.

A set of pairs of lexical and surface characters—written as lexical character–colon–surface character, e.g. *a:a*, *+:0*—constitutes possible mappings. Pairs with no attached rules are applied by default. For all other pairs the attached rules restrict their application to a certain phonological context. Rules function as constraints on the mapping between surface and lexical form of morphs. They are applied in parallel and not one after the other as in generative phonology. Since no ordering of the rules is involved this is a completely declarative way of description.

A rule consists of the following parts:

- The **substitution** indicates the affected character pair.
- **left** and **right context** are regular expressions that define the phonological conditions for the substitution.
- **operators** define the status of the rule: the **context restriction operator** ⇐ makes

the substitution of the lexical character obligatory in the context defined by that rule (other phonological contexts are not affected). The **surface coercion operator** ⇒ restricts the substitution of the lexical character to exactly this context (it may not occur anywhere else). The ⇔ is a combination of the former two, i.e. the substitution must take place in this and only this context. The fourth operator /⇐ states prohibitions, i.e. the substitution may not take place in this context.

The following rule specifies that a lexical morph boundary (indicated by '+') between a *sibilant* on the left side and an *s* on the right side must correspond to surface level *e*. By convention a pair with identical lexical and surface character may be denoted by just a single character. Curly brackets indicate a set of alternatives, square brackets a sequence.

(2.32) *a.* +:e ⇐ {s x z [{s c} h] } : _ s ;

The rule covers some of the cases where *e* is inserted between stem and an inflectional affix starting with *s* (plural, 3rd person, superlative) in English. By default, the morph boundary will map to null, but in the given specific context it maps to *e*. (2.32*a*) makes no statements about other contexts. The following examples demonstrate the application of this rule (vertical bars denote a default pairing, numbers the application of the corresponding rule):

(2.33)

#b l i s s + s #	# f o x + s #	#d i s h + s #	#wa t ch + s #
| | | | | | 1 | |	| | | | 1 | |	| | | | | 1 | |	| | | | | 1 | |
0b l i s s e s 0	0 f o x e s 0	0d i s h e s 0	0wa t ch e s 0

(2.32*a*) does not capture all the cases where *e* epenthesis occurs. For example, the forms *spies*, *shelves*, or *potatoes* are not covered. A more complete rule is:

(2.32) *b.* +:e ⇔ {s x z [{s c} h:h] :v [C y:] [C o]} _ s ;

Rule (2.32*b*) defines all the contexts where '+' maps to *e* (because of the ⇔ operator). It makes use of some additional writing conventions. A colon followed by a character denotes the set of all pairs with that surface character. Accordingly, a character followed by a colon means the set of all pairs with that lexical character. The C stands for the set of English consonants, the V for the vowels. To cope with the *spies* example we need another rule which licenses the mapping from *y* to *i*.

(2.34) y:i ⇔ C _ { +:e [+: e] } ;
V C^+ _ +: C ;

Rule (2.34) specifies two distinct contexts. If either of them is satisfied the substitution must occur, i.e. contexts are OR-connected. The '+' operator in the second context indicates *at least one occurrence* of the preceding sign (accordingly, the operator '*' has the reading *arbitrarily many occurrences*). Jointly with rule (2.35) for the mapping from 'f' to 'v' rule (2.32) also takes care of forms like *shelves* and *potatoes*:

(2.35) f:v ⇐ {el}_+:s;
V_e+:s;

#spy+s#	#toy+s#	#shelf+s#	#wife+s#	#potato+s#
|||21||	|||||||	|||||31||	|||3||||	|||||||1||
0spies0	0toy0s0	0shelves0	0wive0s0	0potatoes0

A given pair of lexical and surface strings can only map if they are of equal length. There is no possibility of omitting or inserting a character in one of the levels. On the other hand, elision and epenthesis are common phonological phenomena. To cope with these, the null character (written as 0) is included in both the surface and the lexical alphabet. The null character is taken to be contained in the surface string for the purpose of mapping lexical to surface string and vice versa but it does not show up in the output or input of the system. Diacritics are mapped to the null character by default. Any other mapping of a diacritic has to be licensed by a rule.

Assumption of the explicit null character is essential for processing. A mapping between a lexical and a surface string presupposes that for every position a character pair exists. This implies that both strings are of equal length (nulls are considered as characters in this respect). Rules can either be directly interpreted or compiled into finite-state transducers. The use of finite-state machinery allows for very efficient implementation. For a more in-depth discussion of implementational aspects consult Chapter 18 or Beesley and Karttunen (2001).

One subtle difference between direct rule interpretation and transducers shows in the repeated application of the same rule to one string. The transducer implicitly extends the phonological context to the whole string. It must therefore explicitly take care of an overlapping of right and left contexts (e.g. in (2.32) the pair *s:s* constitutes both a left and right context). With direct interpretation a new instance of the rule is activated every time the left context is found in the string and overlapping need not be treated explicitly.

2.8.1.1.2 *The continuation lexicon*

A partitioned lexicon of morphs (or words) takes care of word formation by affixation. The lexicon consists of (non-disjunctive) sublexicons, so-called continuation classes. Morphs that can start a word are stored in the so-called *init lexicon*. For every morph, a set of legal continuation classes is specified. This set defines the sublexicons that must be searched for continuations. The whole process is equivalent to stepping through a finite automaton. A successful match can be taken as a move from some state *x* of the automaton to some other state *y*. Lexical entries can be thought of as arcs of the automaton: a sublexicon is a collection of arcs having a common *from* state.

The lexicon in two-level morphology serves two purposes: one is to describe which combinations of morphs are legal words of the language, the other one is to act as a filter whenever a surface word form is to be mapped to a lexical form. Its use for the second task is crucial because otherwise there would be no way to limit the insertion of the null character.

For fast access, lexicons are organized as letter tries (Fredkin 1960). A trie is well suited for an incremental (letter-by-letter) search because at every node in the trie exactly those continuations leading to legal morphs are available. Every node in the trie represents the sequence of characters associated with the path leading to that node. With nodes representing a legal morph their continuation classes are stored. In recognition, search starts at the root of the trie. Each proposed character is matched against the lexicon. Only a legal continuation at that node in the tier may be considered as a possible mapping.

Recent implementations collapse the lexicon and the two-level rules into a single, large transducer, resulting in a very compact and efficient system (cf. Chapter 18).

2.8.1.2 *Related formalisms*

For a more elegant description of phonological (or orthographic) changes affecting sequences of characters Black et al. (1987) propose a rule format consisting of a surface string (LHS for *left-hand side*) and a lexical string (RHS for *right-hand side*) of equal length separated by an operator. Surface-to-lexical rules ($\Rightarrow$) request the existence of a partition of the surface string where each part is the LHS of a rule and the lexical string the concatenation of the corresponding RHSs. Lexical-to-surface rules ($\Leftarrow$) request that any substring of a lexical string which equals an RHS of a rule must correspond to the surface string of the LHS of the same rule. The following rules are equivalent to rule (2.32*a*).

(2.36)

ses $\Rightarrow$ s+s	ses $\Leftarrow$ s+s	shes $\Rightarrow$ sh+s	shes $\Leftarrow$ sh+s
xes $\Rightarrow$ x+s	xes $\Leftarrow$ x+s	zes $\Rightarrow$ z+s	zes $\Leftarrow$ z+s
ches $\Rightarrow$ ch+s	ches $\Leftarrow$ ch+s		

These rules collapse context and substitution into one undistinguishable unit. Instead of regular expressions only strings are allowed. Because surface-to-lexical rules may not overlap, two different changes that happen to occur close to each other must be captured in a single rule. Also, long-distance phenomena like vowel harmony cannot be described in this scheme. As a remedy, Ruessink (1989) reintroduces contexts. Both LHS and RHS may come with a left and right context. They may also be of different length, doing away with the null character. Though Ruessink gives no account of the complexity of his algorithm one can suspect that it is in general less constrained than the original system.

An inherently difficult problem for two-level morphology is the root-and-template morphology of Semitic languages. One solution is the introduction of multi-tape formalisms as first described in the seminal paper by Kay (1987). The best-documented current system is SEMHE (Kiraz 1996), based on Ruessink's formalism with the extension of using three lexical tapes: one for the root, one for the vowel pattern, and one for the template.

Another extension to the formalism is realized in X2MorF (Trost 1992). In the standard system, morphologically motivated phenomena like umlaut must be

described by introducing pseudosegmental material in the lexical level (see section 2.8.3). In X2MorF an additional morphological context is available to describe such phenomena more naturally.

2.8.2 Alternative formalisms

Alternative proposals for morphological systems include so-called paradigmatic morphology (Calder 1989) and the DATR system (Evans and Gazdar 1996). Common to both is the idea of introducing some default mechanism which makes it possible to define a hierarchically structured lexicon where general information is stored at a very high level. This information can be overwritten lower in the hierarchy. Both systems seem to be more concerned with morphosyntax than with morphophonology. It is an open question whether these approaches could somehow be combined with two-level rules.

2.8.3 Examples

Finnish vowels are classified into *back*, *front*, and *neutral*. According to vowel harmony all vowels in a word must be either back or front (disregarding neutral vowels).

(2.37)
```
V = {a, o, u, ä, ö, y, e, i}
Vb = {a, o, u} Vf = {ä, ö, y}
[1] {A:a|O:o|U:u}  ⇒  =:Vb: =: (-Vf)* _ ;
[2] {A:ä|O:ö|U:ü}  ⇒  {#|=:Vf} =: (Vb)* _ ;

#t a i v a s+t A#      #p u h e l i n+t A#      #s y y+t A#
|||||||||1|           |||||||||||1|           ||||||2|
0 t a i v a s 0 t a 0  0 p u h e l i n t A a 0  0 s y y 0 t ä 0
```

The phonological process of final devoicing in *German* works on syllable structure. Voiced consonants in syllable-final position are devoiced, e.g. the root /ra:d/ (wheel) is realized as /ra:t/ in the singular and as /re:də/ in the plural). This phenomenon is not reflected by orthography.

(2.38)
```
Cx:Cy ⇒ _ #:0 ;
        where Cx in (b d g)
              Cy in (p t k) matched;

#l o : b#     #r a : d#     #we : g#     #we : g+e#
|||  1|      |||  1|      |||  1|      |||  ||||
0 l o : p 0   0 r a : t 0   0 we : k 0   0 we : g 0 e 0
```

While the original linguistic motivation behind two-level morphology was generative phonology, and two-level rules were designed to describe morphophonology, the mechanism can also deal with purely morphological phenomena.

German umlaut is used to mark—among other morphosyntactic features—plural.

(2.39) V={a, ä, e, i, o, ö, u, ü, A:a, A:ä, O:o, O:ö, U:u, U:ü}
{A:ä | O:ö | U:ü} ⇒ _ ?* $:0;

All stem vowels eligible for umlaut are realized at the lexical level by a vowel underspecified for the back/front distinction. A pseudo-ending $ triggers the rule application, thus realizing the umlaut. In all other cases the default pairing is used. This way a morphological property is described as a morphophonological process. The ?* signifies zero or more occurrences of anything.

(2.40)

#mUt t e r+$#	#gA r t en+$#	#hOf+$e#
| | 1 | | | | | | |	| | 1 | | | | | | |	| | 1 | | | | |
0müt t e r000	0g ä r t en000	0h ö f00e0

A (simplified) example from *Tagalog* shows how two-level rules can be used to describe reduplication and infixation. Rule [1] (see 2.41 below) captures infix insertion: On the lexical level, the prefix X is assumed. While X is not realized on the surface, it triggers the insertion of *-in-* between initial consonant and following vowel.

(2.41) V={a, i, u, E}
C={p t k b d g m n N s l r w y R}
[1] X:0 ⇒ _ +:0 C 0:i 0:n V:V

#X+p00 i l i#	#X+t00 a h i#
| 1 | | | | | | | |	| 1 | | | | | | | |
000 p i n i l i 0	000 t i n a h i 0

Rules [2] and [3] (2.42) cause the reduplication of the first (open) syllable: the R copies the initial consonant, the E the following vowel. The rules also take care of the case where the infix is inserted as well:

(2.42) [2] R:Cx ⇒ _ (0:i 0:n) E:V +:0 :Cx;
where Cx in (p p:m t t:n k K: N);
[3] E:Vx ⇒ _R:C (0:i 0:n) _ +:0 C Vx;
where Vx in (a i u);

#RE+p i l i#	#RE+t a h i#
| 2 3 | | | | | |	| 2 3 | | | | | |
0p i 0p i l i 0	0t a 0t a h i 0

#X+R00E+p i l i#	#X+R00E+t a h i#
| 1 | 2 | | 3 | | | | | |	| 1 | 2 | | 3 | | | | | |
000 p i n i 0p i l i 0	000 t i n a 0t a h i 0

Further Reading and Relevant Resources

Morphology and Computation (Sproat 1992) gives a concise introduction into morphology with examples from various languages and a good overview of applications of computational morphology. On the methodological side it concentrates on finite-state morphology. *Computational Morphology* (Ritchie et al. 1992) provides a more in-depth description of finite-state morphology but concentrates exclusively on English. An up-to-date description of finite-state technology can be found in Beesley and Karttunen (2001). The *Handbook of Morphology* (Spencer and Zwicky 1998) offers an excellent overview of morphology with examples from diverse languages.

To get some hands-on experience with morpological processing connect to *RXRC Europe* at http://www.rxrc.xerox.com/research/mltt/ and *Lingsoft* at http://www.lingsoft.fi/. A free downloadable version of two-level morphology is available from *SIL* at http://www.sil.org/pckimmo.

References

Beesley, K. R. and L. Karttunen. 2003. *Finite-State Morphology*. Palo Alto, Calif.: CSLI Publications.

Black, A. W., G. D. Ritchie, S. G. Pulman, and G. J. Russell. 1987. 'Formalisms for morphographemic description', *Proceedings of the 3rd Meeting of the European Chapter of the Association for Computational Linguistics* (*ACL '87*) (Copenhagen), 11–18.

Calder, J. 1989. 'Paradigmatic Morphology'. *Proceedings of the 4th Meeting of the European Chapter of the Association for Computational Linguistics* (*ACL '89*) (Manchester), 58–65.

Chomsky, N. and M. Halle. 1968. *The Sound Pattern of English*. New York: Harper and Row.

Evans R. and G. Gazdar. 1996. 'DATR: a language for lexical knowledge representation'. *Computational Linguistics*, 22(2), 167–216.

Fredkin, E. 1960. 'Trie Memory'. *Communications of the ACM*, 3, 490–9.

Fromkin, V. and R. Rodman. 1997. *An Introduction to Language*. 6th edn, Orlando, Fla.: Harcourt, Brace, Jovanovich.

Kay, M. 1987. 'Nonconcatenative finite-state morphology', *Proceedings of the 3rd Meeting of the European Chapter of the Association for Computational Linguistics* (*ACL '87*) (Copenhagen), 2–10.

Kiparsky, P. 1987. *The phonology of reduplication*. Manuscript. Stanford University.

Kiraz, G. A. 1996. 'SEMHE: a generalized two-level system', *Proceedings of the 34th Meeting of the Association for Computational Linguistics* (*ACL '96*) (Los Altos, Calif.), 159–66.

Koskenniemi, K. 1984. 'A general computational model for word-form recognition and production', *Proceedings of the 10th International Conference on Computational Linguistics* (*COLING '84*) (Stanford, Calif.), 178–81.

Nash, D. 1980. *Topics in Warlpiri grammar*. Ph.D. Thesis, MIT, Cambridge, Mass.

Nida, E. 1949. *Morphology: The Descriptive Analysis of Words*. Ann Arbor: University of Michigan Press.

Ritchie, G. D., G. J. Russel, A. W. Black, and S. G. Pulman. 1992. *Computational Morphology*. Cambridge, Mass.: MIT Press.

Rose, S. 2001. 'Triple take: Tigre and the case of internal reduplication'. In R. Hayward, J. Ouhalla, and D. Perrett (eds.), *Studies in Afroasiatic Grammar*. Amsterdam: Benjamins.

Ruessink, H. 1989. *Two-level formalisms*. Working Papers in Natural Language Processing 5, Rijksuniversiteit Utrecht.

Spencer, A. and A. Zwicky (eds.). 1998. *The Handbook of Morphology*. Oxford: Blackwell.

Sproat, R. W. 1992. *Morphology and Computation*. Cambridge, Mass.: MIT Press.

Trost, H. 1992. 'X2MORPH: a morphological component based on augmented two-level morphology'. *Proceedings of the 12th International Joint Conference on Artificial Intelligence*. San Mateo, Calif.: Morgan Kaufmann, 1024–30.

CHAPTER 3

LEXICOGRAPHY

PATRICK HANKS

ABSTRACT

This chapter provides an overview of computational lexicography in two senses: (1) the function of the lexicon in computer programs; and (2) the use of computational techniques in compiling new dictionaries. The chapter begins with the historical background of lexicography. Next, it discusses the particular challenges of using human dictionaries for computational purposes. It goes on to examine the ways that computational techniques have changed the task of compiling new dictionaries; in these sections, special attention is paid to the links between meaning and use. In the chapter's final sections, future directions and sources for further reading are presented.

3.1 INTRODUCTION

An inventory of words is an essential component of programs for a wide variety of natural language processing applications, including information retrieval, machine translation, speech recognition, speech synthesis, and message understanding. Some of these inventories contain information about syntactic patterns and complementations associated with individual lexical items (see Chapter 4); some index the inflected forms of a lemma to the base form (see Chapter 2); some include definitions; some

provide semantic links to an ontology and hierarchies between the various lexical items (see Chapters 13 and 25). Some are derived from existing human-user dictionaries, as discussed below. None are completely comprehensive; none are perfect. Even where a machine-readable lexicon is available, a lot of computational effort may need to go into 'tuning' the lexicon for particular applications. Sometimes, an off-the-peg lexicon is deemed to be more trouble than it is worth, and a required lexicon may be constructed automatically by induction from texts (see Chapter 21).

At the same time, the craft of lexicography has been revolutionized by the introduction of computer technology. On the one hand, new techniques are being used for compiling dictionaries and word lists of various kinds; on the other, new insights are obtained by computational analysis of language in use.

3.1.1 Definitions

In this chapter, two meanings of the term 'computational lexicography' are distinguished:

1. Restructuring and exploiting human dictionaries for computational purposes.
2. Using computational techniques to compile new dictionaries.

The focus is on computational lexicography in English. A comprehensive survey of computational lexicography in all the languages of the world is beyond the scope of this chapter. Lexicography in many of the world's neglected languages is now being undertaken in many research centres; the work is often computer assisted and associated with a machine-readable product.

3.2 Historical Background

Until recently, the only reason anyone ever had for compiling a dictionary was to create an artefact for other human beings to use. Up to the Renaissance, dictionaries were either bilingual tools for use by translators, interpreters, and travellers, or Latin and Greek word lists for students and scholars. As living languages and cultures became more complex, vocabularies expanded and people began to compile dictionaries of 'hard words' in their own language–learned words which ordinary people might not understand. The earliest example in English is Robert Cawdrey's *Table Alphabeticall ... of Hard Usuall Words ... for the Benefit of Gentlewomen and Other Unskillful Per-*

sons (1604). It was not until the eighteenth century that lexicographers set themselves the objective of collecting and defining *all* the words in a language. For English, this culminated in Samuel Johnson's *Dictionary of the English Language* (1755), containing not only definitions but also illustrative citations from 'the best authors'.

Johnson's was the standard dictionary of English until the end of the nineteenth century, but already in 1857 Richard Chenevix Trench presented a paper to the Philological Society in London, 'On some deficiencies in our English dictionaries', in which he described lexicographers as 'the inventory clerks of the language'. This paper played a large part in motivating the Philological Society's *New English Dictionary on Historical Principles*, alias *The Oxford English Dictionary* (1884–1928).

3.2.1 Deficiencies

Many of the deficiencies that characterized nineteenth-century dictionaries still beset lexicography today, though sometimes in new forms, and they are of computational relevance. They arise from problems of both practice and principle. Chief among them are the following.

3.2.1.1 *Omissions and oversights*

It is literally impossible to compile an exhaustive inventory of the vocabulary of a living language. Trench noted many omissions and oversights in the dictionaries of his day, but the creative nature of the lexicon means that every day new words are created ad hoc and, in most but not all cases, immediately discarded. It is impossible for the inventorist to know which neologisms are going to catch on and which not. Murray deliberately omitted the neologism *appendicitis* from the first edition of *OED*. An American dictionary of the 1950s deliberately omitted the slang term *brainwash*. The first edition of *Collins English Dictionary* (1979) omitted *ayatollah*. In their day, each of these terms was considered too obscure, informal, or jargonistic to merit inclusion, though hindsight proved the judgement to be an error. That said, almost all today's machine-readable dictionaries offer a very high degree of coverage of the vocabulary of ordinary non-specialist texts—well over 99.9 per cent of the words (as opposed to the names). Lexical creativity is peripheral, not central, in ordinary discourse.

3.2.1.2 *Coverage of names*

Coverage of names is a perennial problem. Some dictionaries, on principle, do not include any entries for names; for example, they contain an entry for *English* (because it is classified as a word, not a name), but not for *England*. Other dictionaries contain a selection of names that are judged to be culturally relevant, such as *Shakespeare*, *New York*, *Muhammad Ali*, and *China*. Very few brand names and business names are

found in dictionaries: *Hoover* and *Thermos flask* are judged to have become part of the common vocabulary, but no dictionary includes brand names such as *Malteser* or *Pepsi*, whatever their cultural relevance. No dictionary makes any attempt to include all the names found in a daily newspaper. However, names can be just as important as words in decoding text meaning. In Hanks (1997), discussing the role of immediate-context analysis in activating different meanings, I cited an example from the British National Corpus: in the sentence 'Auchinleck checked Rommel' selection of the meaning 'cause to pause' for *check* depends crucially on the military status of the subject and object as generals of opposing armies. If Auchinleck had been Rommel's batman, or a customs inspector, or a doctor, a different sense of *check* would have been activated.

3.2.1.3 *Ghosts*

Ghost words and ghost senses constantly creep in, evading the vigilance of lexicographers despite their best efforts. Crystal (1997: 111) mentions *commemorable* and *liquescency* as examples of words that have probably never been used outside the dictionaries in which they appear. He goes on to cite *Dord*, glossed as 'density', a ghost word that originated in the 1930s as a misreading of the abbreviation *D or d* (i.e. capital or lower-case d), which does indeed mean 'density'.

3.2.1.4 *Differentiating senses*

No generally agreed criteria exist for what counts as a sense, or for how to distinguish one sense from another. In most large dictionaries, it might be said that minor contextual variations are erected into major sense distinctions. In an influential paper, Fillmore (1975) argued against 'checklist theories of meaning', and proposed that words have meaning by virtue of resemblance to a prototype. The same paper also proposed the existence of 'frames' as systems of linguistic choices, drawing on the work of Marvin Minsky (1975) among others. These two proposals have been enormously influential. Wierzbicka (1993) argues that lexicographers should 'seek the invariant', of which (she asserts) there is rarely more than one per word. This, so far, they have failed to do; nor is it certain that it could be done with useful practical results. Nevertheless Wierzbicka's exhortation is a useful antidote to the tendency towards the endless multiplication of entities (or, to put it more kindly, drawing of superfine sense distinctions) that is characteristic of much currently available lexicography.

3.2.2 Usability of the lexicon

In the emergent United States, the indefatigable Noah Webster published his *American Dictionary of the English Language* (1828), a work which paid particular attention

to American English, which was already beginning to differ from standard British English, although its definitions owe more to Johnson than its compiler liked to admit. Johnson, Murray, and Webster all compiled their dictionaries on 'historical principles'. That is, they trace the semantic development of words by putting the oldest meanings first. This is a practice still followed by many modern dictionaries. It is of great value for cultural and literary historians, but at best an unnecessary distraction and at worse a potential source of confusion in computational applications. For purposes of computational linguistics, if word meaning is in question at all, it is more important to have an inventory that says that a *camera* is a device for taking photographs than to know that, before the invention of photography, the word denoted 'a small room' and 'the treasury of the papal curia'.

The earliest comprehensive dictionary to make a serious attempt to put modern meaning first was Funk and Wagnall's (1898). Unfortunately, the great Funk and Wagnall's dictionaries of the early twentieth century no longer exist in any recognizable form. Current American large dictionaries that claim to put modern meanings first are *The Random House Dictionary* (1964, 1996), the second edition of which is available on CD-ROM, and *The American Heritage Dictionary* (1969; 4th edn. 2000). A British counterpart is *Collins English Dictionary* (1979; 4th edn. 1999).

Because they not only put modern meanings first, but also contain fuller syntactic information (including, in some cases, more or less sophisticated indications of **subcategorization** and **selectional preferences**), dictionaries for foreign learners are popular among computational researchers and tool builders. The pioneering work in this class was A. S. Hornby's *Oxford Advanced Learner's Dictionary of Current English* (*OALDCE*; 1948). The sixth edition (2000) has been fully revised, taking account of corpus evidence from the British National Corpus.

3.2.3 Machine-readable dictionaries (MRDs)

Most such dictionaries are available in **machine-readable** form, and research rights can sometimes be negotiated with publishers. To overcome problems of commercial sensitivity, in some cases older editions are licensed. Probably the most widely cited dictionary in computational applications is the *Longman Dictionary of Contemporary English* (*LDOCE*; 1978; http://www.longman-elt.com/dictionaries). The latest edition of *LDOCE* is available on CD-ROM. Like *OALDCE*, it has been revised using evidence from the British National Corpus. It also devotes considerable attention to spoken English. The electronic database of *LDOCE*, offered under specified conditions for NLP research, contains semantic domains and other information not present in the published text.

3.2.4 Corpus-based dictionaries

In 1987, with the publication of the COBUILD dictionary (an acronym for 'Collins Birmingham University International Language Database', 1987, 1995), a new development in lexicography emerged: the **corpus-based dictionary**. The word 'corpus' is nowadays a fashionable buzzword designating any of a wide variety of text collections (see Chapter 24). In the sense most relevant to lexicography, a corpus is a collection in machine-readable form of whole texts or large continuous extracts from texts. Such a collection provides a more *statistically valid* base for computational processing and study of contemporary English than a collection of citations or quotations. A corpus can be used to study words in use, but only indirectly to study word meanings. COBUILD is more intimately connected with its corpus than any other dictionary. It offers a highly interactive and informative website (http://titania.cobuild.collins.co.uk). Unlike the British National Corpus, which maintains its balance by being static, the so-called 'Bank of English' is dynamic: a so-called '**monitor corpus**', constantly growing. At the time of writing it consists of over 330 million words of running text. This provides Collins lexicographers with a magnificent resource for studying new words and meanings.

A recent addition to the stock of major corpus-based dictionaries is the *Cambridge International Dictionary of English* (*CIDE*; 1995; http://dictionary.cambridge.org), which has a number of interesting features, including associated data modules for NLP such as lists of verb complementation patterns, semantic classifications of nouns, and semantic domain categories.

In 1998, Oxford University Press published *The New Oxford Dictionary of English* (*NODE*), a dictionary for native speakers of English (as opposed to foreign learners) which draws both on the citation files of the large historical *Oxford English Dictionary*, collected by traditional methods, and on new corpus resources, in particular the British National Corpus of 100 million words of text. Use of a corpus enables lexicographers to make more confident generalizations about common, everyday meanings, while citation files provide a wealth of quotations to support rare, interesting, new, and unusual words and uses.

The biggest word list in a one-volume English dictionary is to be found in *Chambers English Dictionary*. This magnificent ragbag of curiosities achieves its vaunted 215,000 references by including a great deal of archaic Scottish and other dialect vocabulary (e.g. '**giz** or **jiz** (*Scot*) a wig') and obsolete literary forms (e.g. '**graste** (*Spenser*) *pa p* of *grace*'), of more interest to Scrabble players than to serious computational linguists.

The foregoing paragraphs mention the main 'flagship' dictionaries likely to be of interest to computational linguists. Each of the flagship publications is associated with a family of other lexical reference works, for example thesauri, dictionaries of idioms, dictionaries of phrasal verbs, dictionaries for business English, and smaller derivative works.

Section 3.5 of this chapter discusses corpus-based lexicography in Britain in more detail. No dictionaries based on serious large-scale corpus research have yet been prepared in the United States, although the *American Heritage Dictionary* made some use of the pioneering Brown Corpus of the 1960s (1 million words; see Francis and Kučera 1982), and an American edition of *NODE*, called the *New Oxford American Dictionary* (*NOAD*) was published in 2001. From a lexicographical point of view, a large corpus is an indispensable tool of the trade for serious compilation of paper dictionaries and computational lexicons alike. Studying the patterns of actual usage of words in a balanced and representative selection of texts such as the British National Corpus (www.hcu.ox.ac.uk/BNC; see Aston and Burnard 1998) or the forthcoming American National Corpus (see Ide and Macleod 2001) provides an essential antidote to the distortions created by introspective reporting of the lexicon, typical of older dictionaries.

3.3 Restructuring and Exploiting Human Dictionaries for Computational Purposes

All humans—foreign learners, native speakers, translators, and technical specialists alike—share certain attributes that are not shared by computers. Typically, humans are very tolerant of minor variation, whereas a computer process may be thrown by it. For example, the first edition of the *Oxford English Dictionary* (*OED*) contains innumerable minor variations that the nineteenth century compilers were unaware of or considered unimportant. To take a simple example, 'Shakes.', 'Shak.', and 'Shakesp.' are among the abbreviations used for 'Shakespeare'. When *OED* was prepared for publication in machine-readable form, at first on CD-ROM, and now on line (http://www.oed.com/), the editors spent much time and effort *standardizing* the text in order to ensure that user searches would produce comprehensive results as well as being swift, efficient, and robust. Imposing standardization has been a major concern for making dictionaries **machine tractable**. At the more complex end of the spectrum, it is clearly desirable to impose standardization in definition writing, so that, for example, the definitions for all edible marine fish would be retrievable by searching for a single defining phrase. This involves standardization of innumerable variations such as 'eatable fish', 'strong-tasting fish', 'edible sea fish', 'edible flatfish', 'marine fish with oily flesh', etc. Such tasks present a potentially infinite series of challenges for the

standardizer. Attempts to devise short cuts or automatic procedures using resources such as a machine-readable thesaurus can lead to unfortunate consequences, such as equating the meaning of 'shaking hands' with 'shaking fists'.

Early work in creating MRDs generally involved converting typesetters' tapes into a database format. Unbelievably large quantities of typographical instructions had to be stripped out, leaving just a few that could be converted into logical **field delimiters**. Nowadays, new dictionaries are routinely set up from the outset as structured files or databases, from which typesetters' files are derived. However, the vast size and cost of dictionaries, their long gestation periods, and the great length of their marketing lives mean that there are still quite a few electronic dinosaurs lumbering about, containing valuable information in text but encrusted with typographic details.

The earliest MRD was the computerization at SDC (Systems Development Corporation), of *Webster's 7th New Collegiate Dictionary* (Olney 1967; Revard 1968), which was keyboarded from the printed text. The choice of text still seems surprising, in view of the historical principles which determine the order of definitions in this dictionary and the complete absence of any clues linking meanings to use, other than basic part-of-speech classes. However, the project leaders presumably took the view that one dictionary is as good as any other, or else that the market leader for human use (selling over a million copies a year) must be good for computer applications. Among other things, the SDC group explored word frequencies in definitions, postulating a privileged semantic status for certain frequent terms such as 'substance, cause, thing', and 'kind', akin to the semantic primitives of Wierzbicka and Wilks, or the 'semantic parts of speech' of Jackendoff. Revard later wrote that, in an ideal world, lexicographic definers would 'mark every . . . semantic relation wherever it occurs between senses defined in the dictionary' (Revard 1973).

Among the most comprehensive analyses of a machine-readable dictionary for lexicographic purposes is the work on *LDOCE* carried out under the direction of Yorick Wilks at New Mexico State University, and subsequently the University of Sheffield. The electronic database of *LDOCE* contains information going far beyond what appears in the published text, for example a systematic account of semantic domain. This work is reported in Wilks, Slator, and Guthrie (1996), which also includes a comprehensive survey of other work on making dictionaries machine tractable. An earlier survey volume is Boguraev and Briscoe (1989), a collection of nine essays describing work in the 1980s to extract semantic and syntactic information from dictionaries, in particular *LDOCE*. A more recent collection of relevant papers is Guo (1995).

The information encoded in large lexicons is widely used in algorithms for procedures such as sense coercion for unknown words (see, for example, Pustejovsky 1993) and word-sense disambiguation. Stevenson and Wilks (2001 and this volume, Chapter 13), for example, report a word-sense disambiguation algorithm trained on a large dictionary-based vocabulary, applying principles of preference semantics to a combination of several different knowledge sources, including part-of-speech tag-

ger, shallow parser, semantic domain tagger (using *LDOCE*'s semantic codes), and semantically tagged corpus. It seems inevitable that a large lexicon of known facts is a prerequisite for determining unknown facts in text processing, for example choosing between senses of a word in a dictionary or assigning a semantic role to an unknown word. For computational applications such as these, dictionaries intended for human use are essential but not ideal. They are essential because they provide a reliable inventory. The most striking disadvantages of using currently available human dictionaries for computational purposes are:

- human dictionaries tend to make very fine semantic distinctions, which are not always computationally tractable and which, in many cases, make no practical difference to the interpretation or processing. It is hard for an algorithm to distinguish between an important and a trivial sense distinction;
- different senses of a word are not clearly, explicitly, and systematically associated with different syntagmatic patterns;
- information about comparative frequency of different words and senses is not given. (Recent editions of British learners' dictionaries have begun to do this for words in a broad-brush-stroke impressionistic fashion, but not for senses.)

Despite these drawbacks, a machine-readable version of a human dictionary is a great deal better than nothing, providing an inventory of all the words that are in ordinary conventional use, and a wealth of data that can be mined with successful results, given sufficient ingenuity and patience.

3.4 Dictionary Structure

Dictionaries are more highly structured than almost any other type of text. Nowadays, the norm is to follow the TEI (text-encoding initiative; www.uic.edu/orgs/tei) for SGML- and HTML-compatible mark-up.

The **tag set** for an entry in the *New Oxford Dictionary of English* may be regarded as typical. A simplified version of the basic structure is set out below, although it should be noted that *NODE* uses many additional, optional tags for various different kinds of information. The main tag set, with nesting (embedding) as shown, is as follows:

⟨se⟩ standard entry, *or*
⟨ee⟩ encyclopedic entry, *embedding*:
  ⟨hw⟩ headword
  ⟨pr⟩ pronunciation
  ⟨s1⟩ sense level 1 (part of speech)
    ⟨ps⟩ part of speech

⟨s2 num=n⟩ sense level 2, with number attribute, *embedding*:
⟨df⟩ definition
⟨ms⟩ meaning extension
⟨ex⟩ example of usage (taken from the British National Corpus or the *Oxford English Dictionary* citation files)
⟨et⟩ etymology
⟨drv⟩ derivative form, *embedding*:
⟨ps⟩ part of speech

Additional tags are used for optional and occasional information, for example technical scientific nomenclature, grammatical subcategorization, significant collocations within ⟨ex⟩ examples, and usage notes. This tag set is derived from the even more elaborate tag set designed in the 1980s for the *OED*. Tagged, consistently structured dictionary texts can be searched and processed by algorithms of the kind designed by Tompa (1992) and his colleagues at the University of Waterloo. This software was designed with the computerized *OED* in mind, but it has a much wider range of applicability, to machine-readable texts of all kinds. The two principal components of this software are PAT, a full-text search system offering a powerful range of search options, and LECTOR, a text display facility. PAT allows users to construct combinations of results using Boolean expressions or proximity conditions. Depending on the text structure, search conditions can be specified within certain fields or regions, some of which are predefined, while others may be made up ad hoc by the user. For example, a user may wish to find all definitions containing the word 'structure' in entries for words beginning with R. PAT enables rapid text searches and retrieval within specified fields of specified groups of entries.

3.5 Using Computational Techniques to Compile New Dictionaries

Lexicographers were quick to seize on the benefits of computers in compiling and typesetting new dictionaries. As long ago as 1964, the *Random House Dictionary of the English Language* was set up as an electronic database, so that different technical senses could be dealt with in sets, regardless of alphabetical order, by relevant experts, thus greatly improving the *consistency of treatment*. Clearly, consistency of treatment in a dictionary benefits from compilation of entries for domain-related and semantically related words together as sets, without regard to where in the alphabet they happen to fall. This is now standard practice in the compilation of all new dictionaries

(as opposed to revised editions and derivative or shortened versions, which usually proceed alphabetically).

3.5.1 Challenges of corpus-based lexicography

Corpus-based lexicography raised a whole new raft of issues, affecting the selection, arrangement, and definition of the lexical inventory. For example, there may be plentiful evidence for a verbal adjective, e.g. *extenuating*, while the base form (*extenuate*) is rare or non-existent. Should there be an entry for the base form, the verbal adjective, or both? Should the idealized lemma or paradigm set always be allowed to prevail over observed data?

The evidence of a large general corpus can help to identify the most common modern meaning of a word, but it must be treated with caution. Frequency alone is not enough. Corpus lexicographers also need to look at the **distribution**: does the word occur in many different texts, only in a particular domain, or only in a single author? For an idiosyncrasy, even if repeated many times, is still an idiosyncrasy.

Another trap is the **failure-to-find fallacy**. Failure to find a particular word or sense in a corpus does not mean that that sense does not exist. It may exist in a register or domain that is inadequately represented in the corpus. On the other hand, it might be argued that a word, phrase, or sense that does not occur in a balanced corpus of 100 million words (let alone 300 or 400 million words), containing a broad selection of text types, cannot be very important—or, rather, can only be of importance in a highly restricted domain.

Corpus lexicographers invoke criteria such as **generalizability** to identify the '**core meaning**' of a word. So, for example, the expression *to shake one's head* is far more common in the British National Corpus than *to shake a physical object*, but the latter sense is still identified as the core meaning and placed first because the range of possible direct objects is so much wider. Core meanings have wider ranges of normal phraseology than derivative, pragmatic, metaphoric, and idiomatic senses.

Identifying the 'literal' modern meaning of a word is often far from straightforward. A sense whose status is that of a conventionalized metaphor may be more common than the so-called literal sense. Literal meanings are constantly on the move: today's metaphor may be tomorrow's literal meaning. Thus, *torrents of abuse* and *torrents of verbiage* may be more common in a large corpus of modern English than *torrents* denoting violently rushing mountain streams, but most English speakers would agree that the latter is nevertheless the literal meaning. It is often difficult to know how far to modify historical principles in describing modern English. For example, the oldest meaning of *check* is the chess sense, closely followed by 'cause to pause or suffer a setback', originally a metaphor based on chess. From this developed the 'inspect' sense, which is by far the most frequent sense today. Which of these senses should be classified as the literal meaning of the verb *check*?

3.5.2 Corpus-based revision

In the 1990s, British dictionary publishers, especially publishers of foreign learners' dictionaries, invested substantially in revising their dictionaries to conform better with corpus evidence, both for the word list and for the meaning and use of words. Corpus-driven revision can involve wholesale rewriting and restructuring of definitions, seeking levels of generalization that conform with the evidence. This in turn might affect the view of semantic hierarchies or ontologies derived from or associated with machine-readable dictionaries, though to the best of my knowledge no systematic comparison has been carried out. For more on ontologies, see Chapter 25.

3.5.3 WordNet

A revolutionary development of the 1990s was WordNet (see Fellbaum 1998; http://www.cogsci.princeton.edu/~wn/), an on-line reference system combining the design of a dictionary and a thesaurus with the rich potential of an ontological database. Instead of being arranged in alphabetical order, words are stored in a database with hierarchical properties and links, such that *oak* and *ash* are subsumed under *tree*. Fourteen different senses of *hand* are distinguished, each with its own set of links. WordNet's design was inspired by psycholinguistic theories of human lexical memory. English nouns, verbs, adjectives, and adverbs are organized into synonym sets, each representing one underlying lexical concept. Different relations link the synonym sets.

It has to be said, however, that, while WordNet's design is new and ground-breaking, its lexicography is often disappointing, owing virtually nothing to corpus linguistics and far too much to traditional dictionaries on historical principles. So, for example, the first sense for the verb *fan* is glossed as 'strike out a batter, in baseball' and sense 4 is 'separate from chaff; of grain'. It cannot be claimed that either of these senses is central to general contemporary usage. The gloss at sense 3, 'agitate the air', is technically deficient, in that it fails to indicate that this is normally a transitive verb with a selectional preference for a direct object denoting a person (or a person's face or body). A systematic revised edition of WordNet, taking account of current advances in lexicographic methodology and resources, would be highly desirable. The present situation, in which different groups of researchers make their own adjustments on a piecemeal basis, is far from satisfactory.

In 1996, a European initiative, EuroWordNet, was set up to build a semantic net linking other European languages to the original English WordNet. EuroWordNet aims to be a standard for the semantic tagging of texts and an interlingua for multilingual systems of information retrieval and machine translation. The user can look up a term in Dutch and get synonyms in English, Spanish, or Italian. EuroWordNet could well turn out to be a strategically significant language tool in enabling everyday com-

munication and commerce to take place in the diverse languages of Europe. It must be noted, however, that the theoretical assumptions underlying WordNet are not universally accepted. The psychological reality of hierarchically organized ontologies is controversial. Many words, inconveniently, do not fit neatly into an ontological hierarchy at all, while others fit equally well (or badly) at many places.

The single most important feature of the WordNet projects, like that of many more traditional research projects, is **coverage**. Unlike most other institutionally funded research projects, WordNet says something about everything. And, unlike commercial projects, it is free.

For a more detailed account of WordNet see Chapter 25.

3.6 Linking Meaning and Use

A serious problem for computer applications is that dictionaries compiled for human users focus on giving lists of meanings for each entry, without saying much about how one meaning may be distinguished from another in text. They assume a decoding application for the dictionary, in which ordinary human common sense can be invoked to pick out the relevant meaning from a list of competing choices. Computers, on the other hand, do not have common sense. Many computer applications need to know how words are used and, ideally, what textual clues distinguish one sense from another. On this subject, dictionaries are largely silent. Learners' dictionaries offer syntactic patterns, but these are at a clausal level, without any more delicate distinction between different semantic classes of direct object.

Choueka and Luisgnan (1985) were among the first to describe the essentials of choosing an appropriate meaning by reference to the immediate **co-text**. This is a technique that has been widely employed and developed since, but is still a subject on which further research is needed. Part of the problem is distinguishing signal from noise, while another is **lexical variability**. It is clear that there are statistically significant associations between words (see Church and Hanks 1989; Church et al. 1994), but it is not easy to see how to establish that, for purposes of choosing the right sense of *shake*, *earthquake* and *explosion* may be equated, while *hand* and *fist* may not. Corpus lexicographers often cite the words of J. R. Firth (1957): 'You shall know a word by the company it keeps.' Much modern research is devoted to finding out exactly what company our words do keep. This work is still in its infancy. Establishing the **norms and variations** of phraseology and **collocation** in a language will continue to be important components of many lexicographic projects for years to come. In 1999 a European Society for Phraseology (Europhras; www.europhras.unizh.ch) was

founded, with the specific objective of promoting the study of phraseology, with relevant results for future lexicography.

COBUILD's innovative defining style expresses links between meaning and use by encoding the target word in its most typical phraseology (e.g. 'when a horse *gallops*, it runs very fast so that all four legs are off the ground at the same time') as the first part of the definition (see Hanks 1987). COBUILD does this impressionistically and informally, in a way designed for human users (foreign learners), not computers, but in principle a project to express similar information in a formal, computer-tractable, way is entirely conceivable. The editor-in-chief of COBUILD, John Sinclair, briefed his editorial team: 'Every distinction in meaning is associated with a distinction in form.' A great deal of research is still required to determine exactly what counts as a distinction in meaning, what counts as a distinction in form, and what is the nature of the association. The immediate local co-text of a word is often but not always sufficient to determine which aspects of the word's meaning are active in that text. For further discussion, see Hanks (1996, 2000).

The Japanese Electronic Dictionary Research Institute (http://www.iijnet.or.jp/edr/) has developed a series of eleven linked on-line dictionaries for advanced processing of natural language by computers. Subdictionaries include a concept dictionary, word dictionaries, and bilingual dictionaries (English–Japanese). The *EDR Electronic Dictionary* is aimed at establishing an infrastructure for knowledge information processing.

3.7 Exploring the Future

Until recently, innovation has been very much the exception rather than the rule in lexicography. Lexicography characteristically aims at breadth, not depth, and most of the lexicographic projects funded by the publishing industry have been required, for commercial reasons, to reach a very wide popular audience. Unlike most researchers, teams of lexicographers are obliged by the nature of their undertaking to say something about everything, even if they have nothing to say. These and other constraints mean that the style and presentation of most dictionaries tends to be very conservative, reflecting eighteenth-century concepts of meaning and definition for example. The main exception to this rule among published dictionaries is COBUILD.

In recent years, a number of research projects have explored possible new approaches to capturing, explaining, defining, or processing word meaning and use. Such studies may not yet cover the entire vocabulary comprehensively, but they have begun to explore new methodologies based on recent research in philosophy of lan-

guage, cognitive science, computational linguistics, and other fields, along with new resources, in particular corpora. They point the way towards more comprehensive future developments. Some of the most important of these projects are mentioned in this section.

The European Community's Research and Development Service (www.cordis.lu/) provides information on research projects funded by the EC. Of particular relevance was the Information Technologies programme of 1994–8 (named *Esprit*; see http://www.cordis.lu/esprit/src/). This sought, with an emphasis on commercial relevance, to favour research in the languages of Central Europe, the Baltic States, the Mediterranean region, and the states of the former Soviet Union, designed to bring the information society to everyone, including speakers of minority languages.

A major theme in the EC's 'Fifth Framework' (1998–2002) is the development of 'Information Society technology' (IST; www.cordis.lu/ist/). There was disappointingly little provision for lexicographic research in this framework. Probably the most important such project is Defi at the University of Liège, which explores how to use the immediate context of a word in a text to select the right translation.

In the 'Fourth Framework' lexicographically relevant projects were funded such as DELIS, COMPASS, SPARKLE, and EAGLES, all of which are described on the Cordis website.

HECTOR. The HECTOR project (Atkins 1993; Hanks 1994) was a fifteen-month experimental collaboration between a computing research laboratory (the Systems Research Center of Digital Equipment Corporation) and a publisher (Oxford University Press). Approximately 1,400 words were studied in detail. Among the objectives were:

1. To provide a 'guinea-pig' project for software engineers developing large-scale corpus-handling software, search engines, graphical user interfaces, pointers, and writers' tools.
2. To categorize exhaustively, in terms of dictionary senses, all uses of the target words of a given general corpus.
3. To explore whether existing Oxford dictionaries such as the *Concise Oxford Dictionary* (*COD*), 8th edn. (1990) would benefit from corpus analysis or whether a new kind of dictionary was needed.
4. To develop the methodology of corpus analysis for lexicographical purposes.

Objectives (1), (3), and (4) were fulfilled. The SRC scientists went on to develop new search-engine technology, and the Oxford lexicographers went on to develop *NODE*, an entirely fresh look at the English language, which drew heavily on the British National Corpus for the organization of its entries, and from which *COD*, 10th edn, was derived. Interestingly, however, objective (2) proved to be much harder to fulfil. It was simply impossible to map *all* uses of the selected target words in the 18-million-word HECTOR corpus (a prototype of the 100-million-word British National

Corpus) onto the senses in the 8th edition of *COD*. This was not because *COD* was a bad dictionary, but rather because it focused on unusual senses rather than everyday ones and (like most dictionaries) made sense distinctions without specifying a decision procedure for distinguishing them. Even after a decision was made to create an entirely new, corpus-driven, customized HECTOR dictionary for the target words, the problems did not go away. For some perfectly normal-seeming uses of everyday words, it was impossible to decide between two or three dictionary senses, and yet there was no motivation to add a new sense. Rather, these uses seemed to activate senses only partially, or activated different components of two or three senses, rather than the whole of any one sense. This might suggest that the whole theoretical concept of word meaning needs to be reconsidered. For computational purposes, perhaps the folk notion of 'word meaning' needs to be replaced with something more practical from a processing point of view, e.g. the **presuppositions** and **entailments** associated with words in their **normal phraseological contexts.** It is all too easy for lexicographers to select examples that suit their purposes and to gloss over those that do not. The requirement to account for *all* corpus uses of the target words, including the ones that did not fit neatly anywhere, was a valuable discipline, unique to the HECTOR project.

Unfortunately, it never became possible to edit HECTOR for publication or to impose internal consistency or completeness on its entries. Nevertheless, in 1999, they were used as a benchmark for the Senseval project in word sense disambiguation (Kilgarriff 1998). For a fuller discussion of the word-sense disambiguation problem, see Chapter 13.

The generative lexicon. Recent work by Pustejovsky (1995) and his followers (see, e.g., Bouillon and Kanzaki 2001) on the '**generative lexicon**' addresses the problem of the multiplicity of word meaning: how we are able to generate an infinite number of senses for individual words given only finite means. Generative lexical theory deals, among other things, with the creative use of words in novel contexts, in a way that is simply beyond the scope of possibility in a finite, 'sense-enumerative' dictionary, but which seems ideally suited for dynamic processing by computer program. According to Pustejovsky, there are three aspects of the lexical structure of a word that impact the mapping of semantic information to syntax: an *argument structure*, an *event structure*, and a so-called *qualia structure*. Qualia for entities are:

formal: the basic category that distinguishes the term within a larger domain
constitutive: the relation between an object and its constituent parts
telic: its purpose and function
agentive: factors involved in its origin

In the generative lexicon, semantic types can constrain the meaning of other words. It has long been recognized that, for example, the verb *eat* imposes the interpretation [[FOOD]] on its direct object, regardless of how that direct object is actually realized.

Pustejovsky goes further, and shows how the sentence 'she enjoyed her coffee' entails an event type (namely *drinking*) and 'he enjoyed his book' entails a different event type (namely *reading*). The verb *enjoy* requires an event semantically as its direct object, so even though *coffee* is not an event, the semantics of a **prototypical** event type (*what do you do with coffee?—drink it*) are coerced by the verb *enjoy*. Different practical aspects of the implications of generative lexicon theory are currently being implemented by a team led by Pustejovsky himself and by others in the USA and elsewhere. The generative lexicon is no different from any other lexicographical project in this regard at least: coverage is key. It tries to say something about everything.

Framenet. Fillmore and Atkins (1992) describe another, equally exciting development in lexicon theory, subsequently put into development as Framenet (http://www.icsi.berkeley.edu/~framenet/). Framenet started by analysing verbs with similar meanings (e.g. verbs of movement), and showing how they are distinguished by the different semantic **case roles** of their arguments. Framenet is grounded in the theory of Frame Semantics, which starts with the assumption that in order to understand the meanings of the words in a language we must first have knowledge of the conceptual structures, or **semantic frames**, which provide the background and motivation for their existence in the language and for their use in discourse. Framenet is corpus-based and contrastive (e.g. it asks precisely what semantic features distinguish *creeping* from *crawling*). Its entries provide information, for each sense, about frame membership and the syntactic means by which each Frame Element is realized in the word's surrounding context. These entries summarize, as **valency patterns**, the range of combinatorial possibilities as attested in the corpus. From the point of view of natural language processing, developments such as Framenet and the generative lexicon seem to be the culmination of research in computational lexicography at the beginning of the twenty-first century. The potential for practical applications seems limitless. It is very much to be hoped that Framenet will be implemented comprehensively for the whole English lexicon (with the possible exception of domain-specific jargon), with resultant tools linking word senses to textual phraseology in a robust enough way to reduce the amount of lexical tuning needed to make a lexicon suitable for a wide variety of NLP applications.

Further Reading and Relevant Resources

Useful websites are Robert L. Beard's index of on-line dictionaries and multilingual resources (http://www.yourdictionary.com) and the Omnilex site (http://www.omnilex.com).

For European language resources, two associations are particularly relevant: ELRA (European Language Resources Association; www.icp.grenet.fr/ELRA) and TELRI

(Trans-European Language Resources Infrastructure; www.telri.de). ELRA, based in Luxembourg, promotes the creation and distribution of language resources for research, such as databases of recorded speech, text corpora, terminology collections, lexicons, and grammars. TELRI administers a research archive of computational tools and resources called TRACTOR. Both these associations organize seminars on language resources and corpus research. TELRI runs an annual series of seminars in different locations, with a focus on corpus research; ELRA runs workshops and LREC, a biennial conference on language resources and evaluation.

The most useful readings in computational lexicography are to be found in the proceedings of conferences and in specialist journals.

The Waterloo-OED conference: annually from 1984 to 1994, organized jointly by Oxford University Press and the University of Waterloo Centre for the New *OED* and Text Research, headed by Frank Tompa (Waterloo, Ontario, Canada N2L 3G1). The Proceedings contain accounts of most major developments in computational lexicography in this period, when seminal developments were taking place.

Complex: annual conference organized by the Hungarian Research Institute for Linguistics, Budapest (http://www.nytud.hu/). Proceedings edited by Franz Kiefer, Gabor Kiss, and Julia Pajsz, with many relevant papers.

Euralex: biennial conference of the European Association for Lexicography (www.ims.uni-stuttgart.de/euralex/). Proceedings contain occasional reports on significant computational developments.

International Journal of Lexicography (ed. R. Ilson (to 1997), A. Cowie (from 1998); Oxford University Press; www3.oup.co.uk/lexico/), quarterly. Occasional articles of computational relevance.

Dictionaries: the Journal of the Dictionary Society of North America (ed. William S. Chisholm (to 1999), M. Adams (from 2000); polyglot.lss.wisc.edu/dsna/); annual. Until recently, disappointingly few articles have been of computational relevance.

Other relevant collections of essays include those in Zernik (1991) and Atkins and Zampolli (1994).

The Oxford Text Archive (http://ota.ahds.ac.uk/) and the Linguistic Data Consortium at the University of Pennsylvania (http://www.ldc.upenn.edu/) both hold copies of a variety of machine-readable dictionaries, which are available for research use under specified conditions.

Some dictionary publishers are willing to make machine-readable versions of their dictionaries available for bona fide academic research, though great tenacity and diplomatic skill may be required to achieve agreement and delivery. Publishers' sensitivity about protecting commercial rights in their colossal, high-risk investments, along with the fact that negotiating the free gift of their products is not always among their highest priorities, can be perceived, usually erroneously, as hostility to research.

The *Oxford English Dictionary* is available on CD-ROM. The third edition is currently in preparation and has recently become available as work in progress on line through certain sites (http://www.oed.com/). This magnificent historical monument

is a cultural keystone for the historical study of the English language. That does not, however, necessarily mean that it is suitable as a tool or benchmark for processing word meaning or distinguishing word senses computationally in modern English.

A broad overview of lexicography in English, including an evaluation of the impact of corpus evidence, may be found in Landau (2001).

With regard to lexicography in French, mention should be made of the *Trésor de la langue française*, with 114.7 million words of text and over 400,000 dictionary entries. This is now available on-line thanks to the ARTFL collection (American Research on the Treasury of the French Language; www.lib.uchicago.edu/efts/ARTFL) at the University of Chicago, in cooperation with Analyses et Traitements Informatiques du Lexique Français (ATILF) of the Centre National de la Recherche Scientifique (CNRS) in France. ARTFL also makes available on-line other important and historic French reference works, including Denis Diderot's massive *Encyclopédie* (1751–72).

Acknowledgements

I am grateful for discussion and comments on earlier drafts from Ken Litkowski, Yorick Wilks, and Mark Stevenson. And special thanks to Elaine Mar for help with finalization.

References

Amsler, R. A. and J. S. White. 1979. *Development of a Computational Methodology for Deriving Natural Language Semantic Structures via Analysis of Machine-Readable Dictionaries*. Technical Report MCS77-01315, National Science Foundation. Washington.

Apresyan, Y., I. Mel'čuk, and A. K. Zholkowsky. 1969. 'Semantics and Lexicography: towards a new type of unilingual dictionary'. In F. Kiefer (ed.), *Studies in Syntax and Semantics*. Dordrecht: D. Reidel.

Aston, G. and L. Burnard. 1988. *The BNC Handbook*. Edinburgh: Edinburgh University Press.

Atkins, B. T. S. 1993. 'Tools for computer-aided lexicography: the Hector project'. In *Papers in Computational Lexicography: COMPLEX '93*. Budapest: Research Institute for Linguistics, Hungarian Academy of Sciences.

——J. Kegl, and B. Levin. 1988. 'Anatomy of a verb entry'. *International Journal of Lexicography*, 1(2), 84–126.

——and B. Levin. 1991. 'Admitting impediments'. In U. Zernik (ed.), *Lexical Acquisition: Exploiting On-Line Resources to Build a Lexicon*. Hillsdale, NJ: Lawrence Erlbaum Associates.

——and A. Zampolli (eds.). 1994. *Computational Approaches to the Lexicon*. New York: Oxford University Press.

Boguraev, B. and T. Briscoe. 1989. *Computational Lexicography for Natural Language Processing*. London: Longman.

Bouillon, P. and K. Kanzaki (eds.). 2001. *Proceedings of the 1st International Workshop on Generative Approaches to the Lexicon*. Geneva, Switzerland.