One in the bush

Low-density language technology

Lars Borin

One in the bush
Low-density language technology[*]

Lars Borin

The old quip attributed to Uriel Weinreich, that a language is a dialect with an army and a navy, is being replaced in these progressive days: a language is a dialect with a dictionary, grammar, parser and a multi-million-word corpus of texts – and they'd better all be computer tractable. When you've got all of those, get yourself a speech database, and your language will be poised to compete on terms of equality in the new Information Society. (Ostler n.d.)

Abstract

Language technology is the common name for a spectrum of technologies, tools, algorithms, etc., which enable computers to deal with human language in all its manifestations – speech, writing and sign – as an *open-ended system*, rather than merely as a closed set of linguistic products. Recently, there has been a good deal of concern about the availability of language technology resources for other languages than English and a few others, and especially for lesser-known languages. This report gives a short introduction to language technology and an overview over language technology resources that have been discussed in connection with lesser-known languages.

Proposed methods for the automatic acquisition of linguistic knowledge by computer (so-called *machine learning*) potentially allow for the rapid creation of language technology resources with minimal human work, which if realized would be of great help in the case of many lesser-known languages. However, current machine learning methods – like language technology in general – have arguably been shaped by the typological and other traits of the most explored language, namely English, which is in many respects an atypical language from a linguistic point of view. Research aiming at the development of methods for the rapid creation of language technology resources for lesser-known languages would be doubly beneficial. We would get a better understanding of the generality or language-specificity of these methods, and at the same time, we might conceivably get the embryo of some language technology resources for some lesser-known languages.

---

[*] This report constitutes a longer and more detailed version of a book chapter published earlier as Borin 2006, and it existed more or less in the present form in early 2004 (see also Borin 2009). The title comes from an analogy between the proverb "A bird in the hand is worth two in the bush" and the differences between knowledge-rich – based on hand-crafted grammars (a bird in the hand) – and knowledge-poor – based on machine-learning (two birds in the bush) – methods for building language resources. In the case of many – arguably most – of the world's languages, we do not have a bird in the hand, but it is also doubtful whether we could find more than at most one bird in the bush. The Swedish version of the proverb contrasts a bird in the hand with ten birds in the forest, an even better analogy to the differences between the two approaches, at least at the present state of the art.

1.   Introduction

According to Nicholas Ostler, a language will not get by in the world of today unless it is equipped with a "parser and a multi-million-word corpus of texts" (see the quote above).[1] The parser, and also arguably the computer tractable dictionary and grammar and the text corpus, are examples of *language technology*. Ostler's statement reflects the fact that, recently, there has been a good deal of concern about the availability of language technology resources for other languages than English and a few others.[2] This in turn is part of a larger concern about diminishing linguistic diversity and what can be done to counter this trend (see the papers in Saxena and Borin 2006).

To those outside the field it may not always be completely clear what language technology is all about. The term itself is fairly recent,[3] and there seems to be a tendency (in itself entirely natural and understandable) to refer to any computer application which deals with language in any form as "language technology". Thus, a multimedia program for language learning may be referred to as a piece of "language technology" simply because the purpose of the program is that of assisting language learning.[4] This usage is misleading, however, and should be avoided. The essence of language technology lies not in the circumstance *that* it deals with language in some way – many kinds of technology do that, including ordinary telephones – but in *how* it deals with language. Language technology is the common name for a spectrum of technologies, tools, algorithms, etc., which enable computers to deal with human language in all its manifestations – speech, writing and sign – as an *open-ended system*, rather than merely as a closed set of linguistic products (such as predetermined answers to vocabulary questions or fixed replies retrieved on the basis of of combinations of keywords found in the input). In other words, language technology applications deal with Saussurean *langue* rather than *parole*, in a way which mirrors linguistically statable regularities in a faithful manner, e.g. by not listing such linguistic items which are best described by rules.[5]

---

[1] Actually, it was Max Weinreich (not Uriel) who coined the aphorism that Ostler refers to in the quote.

[2] Mainly western European languages, but also Japanese and Chinese.

[3] A largely synonymous term, with a somewhat longer history of use, is *language engineering*.

[4] On the other hand, it *may* be a piece of language technology, but you have to know or to be able to infer something about its internal workings in order to be able to state this for a fact. In fact, it is my firm belief that language technology will make, e.g., multimedia programs for language learning or language revitalization (such as the *Taitaduhaan* CD-ROM for Western Mono described by Kroskrity and Reynolds 2001) better than the ones produced at present, which as a rule do not use any kind of language technology.

[5] Another way of saying this is that the knowledge of language encoded in language technology applications is *generative* in the same sense that this word is used in,

Language technology is also a kind of information and communic-
ation technology (ICT), however, and it is obviously potentially
applicable whenever and wherever people interact with machines, but
also, and perhaps less obviously, in many cases where people interact
with other people, in the form of various assistive technologies, e.g.
online machine translation, speech-to-text applications for use with
text telephones, etc. Historically, language technology includes the
older fields of computational linguistics/ natural language processing
and speech technology. Hence, language technology is a strongly
interdisciplinary field, with linguistics and computer science being the
traditional ingredients of the more written language-oriented comput-
ational linguistics/natural language processing branch, whereas
phonetics and electrical engineering/ signal processing make up the
speech technology branch of the field. The potential for other cross-
disciplinary combinations is immense, considering that there is hardly
any branch of scientific inquiry (especially in the humanities and
social sciences) which does not at some level involve language.

2.   Language technology resources for less prevalent languages

In the terminology of language technology, a *resource* is the term used
about a collection of data or a piece of software, called *data resources*
and *processing resources* (or *algorithmic resources*), respectively,
depending on whether they are best thought of as static, declarative
knowledge, or dynamic, processual, computer program-like know-
ledge. The data resources are often called *linguistic resources*, since
they consist of samples of language (texts, recordings, etc.) or formal-
ized knowledge about language (dictionaries, grammars, etc.). The
processing resources use linguistic resources in order to analyze
linguistic input or produce linguistic output for some purpose, e.g. for
translating from one language to another, for finding grammatical
errors in text, etc. This compartmentalization into linguistic and proc-

---

e.g., "generative grammar", i.e. they model an infinite set of linguistic objects by fi-
nite means, although in language technology part of the knowledge may be probabil-
istic rather than categorical in nature. Even in those cases you can still often "recon-
struct" some kind of generative system, with probabilities added to, say, the symbols
and productions in a formal grammar. Note that the requirement that language tech-
nology mirror linguistically statable regularities in a faithful manner is best under-
stood as pertaining to linguistic *behavior* and the observable *products* of this behavi-
or – i.e., "performance", in the broad sense of the word – rather than to some system
of linguistic rules supposedly underlying and accounting for this behavior in hu-
mans. In other words, most language technology work is theoretically uncommitted
as to the nature and shape of (human) underlying linguistic knowledge, while still
requiring that regular linguistic behavior (and its products) be captured by law-like
generalizations and mechanisms in language technology applications. Note also that
the common demand placed upon language technology applications, that they be *ro-
bust*, tallies perfectly with this construal of the ontological status of the linguistic
knowledge embodied in such applications.

essing resources – the notion that grammatical generalizations about a particular language should be stated separately from whatever knowledge is needed for processing language[6] – means that in the ideal case only the linguistic resources need to be changed, but not the processor, when we wish to adapt a particular kind of language technology application to a new language.

What, then, are these linguistic and processing resources, and how do you go about creating them for a language which does not have them? In the next few sections below, I will describe a number of resources that have been mentioned in this connection. As my main point of departure for the descriptions I will use the very useful overview made by Kepa Sarasola, a well-known Basque language technology researcher, for the workshop on "Developing language resources for minority languages: reusability and strategic priorities" at the *Second international conference on language resources and evaluation* (Sarasola 2000).

More recently, the US *Linguistic Data Consortium* (LDC) has undertaken a "Low Density Languages" (LoDL) survey, where they have surveyed the largest (in terms of number of native speakers) 300 languages of the world with respect to how well they are equipped with language technology and prerequisite resources (Strassel et al. 2003; see also Borin 2009).[7] They have also made an in-depth survey of 8 of these languages: Bengali, Chechen, Hindi, Panjabi, Tagalog, Tamil, Tigrinya, and Uzbek. The content of sections 2.1 – 2.6 below mainly represents an adaptation of the information in Sarasola's paper, but the data from the LDC LoDL survey have been added to it.[8] Since some of the language technology terminology used in the following sections will most certainly be unfamiliar to readers coming from a background in general linguistics, I will endeavor to explain such terms when they first appear. A summary listing in compact format of all the resources and tools discussed in sections 2.1 – 2.6 can be found in the appendix to this report.

2.1   Prerequisites

The LDC LoDL survey criteria implicitly identify some necessary prerequisites for the (rapid) creation of language technology resources

[6] This is similar to – but not exactly the same as – the distinction made in general linguistics between a grammar formalism (sometimes somewhat high-handedly referred to as a "theory of grammar" or even "theory of language"), and a grammar of a particular language expressed using that formalism.

[7] Some large languages are missing from this survey, notably English, but also German, French, and some others; presumably these languages are considered high-density languages by default.

[8] The LDC survey criteria – which deal mainly with prerequisites, while Sarasola's list is about language technology resources proper (thus the two lists are largely complementary) – are given in square brackets.

for a language, at least using the current state of the art of language technology, which arguably is biased toward English-like linguistic systems (see section 5 below). The LDC LoDL survey criteria fall in the category of prerequisites:

- [language written]
- [standard digital encoding]
- [words separated in writing]
- [simple orthography]
- [sentence punctuation]
- [simple morphology]
- [(existence of) dictionary]
- [(existence of) newspaper]
- [(existence of) Bible (translation)]

It is quite clear from this list that basic literacy is seen as a necessary component of language technology, meaning that not only should there be a standard orthography for the language, but also that the language should actually be used in writing on a regular basis. However, this is not a logical requirement, only a pragmatic one. Current language technology was developed within a written language framework, and consequently it fits most comfortably with well-developed literacy. It is a very interesting question – but well beyond the scope of this report – to consider whether a purely or predominantly oral language technology would be feasible.[9]

Current language technology tools are also aided by certain linguistic and orthographic characteristics, which we can sum up as those that tend to formally delimit and distinguish units of (linguistic) interest, namely lexical word and sentence sized units (at least the criteria "words separated in writing", "simple orthography", "sentence punctuation", "simple morphology"; see also section 5 below).

2.2   Foundations

In this category we find basic linguistic resources which are necessary foundations for all kinds of language technology applications for a particular language. Both more specific digital resources and general linguistic descriptions for human consumption are useful here.

- Corpus: collections of raw text (untagged) [100 000 words of news text]
- Text corpus proper (untagged)
- [100 000 words of parallel text]

---

[9] In the same way that literacy is not a necessary requirement for using e.g. digital telephony, you could make a case for a purely speech-based kind of language technology.

- Lexicon: Raw lists of forms, lemmas, and affixes
- Machine-readable dictionaries (monolingual, bilingual, thesaurus, other)
  [10 000 word translation dictionary]
- Morphology: Description and formalization of morphological phenomena
- Speech databases (collections of digitized speech)
- Formal description and dictionaries of units for speech synthesis

The most central resources in this connection are the *text corpus* proper and the *speech database*. The former is generally the basis for all kinds of language technology dealing with written language, whereas the latter forms the empirical basis for speech technology applications. However, in the remainder of this report, I will not say anything more about speech resources, not because they are unimportant, but because my competence lies elsewhere. Instead, I will mainly address the issues of corpus collection and the creation of tools for linguistic annotation of corpora.

Although they make up important basic resources for language technology, not all uses of text corpora fall within its purview. In fact, most work conducted under the heading of *corpus linguistics* (see McEnery and Wilson 2001 for a general introduction) has very little to do with language technology. Rather, this is traditional descriptive linguistics using fairly simple computer tools for searching, counting, and reordering large amounts of normally unannotated (see below) digital text. This kind of corpus use is relevant anyhow for the purposes of this discussion, since it provides important input for the descriptive grammars and above all the dictionaries mentioned among the foundational resources.

A text corpus proper, as opposed to a 'mere' collection of text – e.g. newstext downloaded from the WWW – is a selection of text intended to be representative for some particular purpose. E.g. a so-called balanced corpus of (published) written language is compiled from the main genres/text types of published written language, in roughly the correct proportions, on the assumption that linguistic investigations of this material will yield results that generalize to written language in general. Compiling a proper text corpus entails a much greater amount of work than merely collecting any kind of text that you can lay your hands on, especially where other text types than newstext are difficult or impossible to acquire in electronic form. A *parallel corpus* is a bi- or multilingual text material containing original texts in one language and their translations into another language or other languages. Often, parallel corpora are *aligned*, meaning that corresponding units (sentences, phrases, even words) from the different language versions are explicitly linked together (see Borin 2002a).

Raw text collections and – even better – proper basic corpora are arguably today the single most valuable resources for language technology development, however, especially for those researchers who see automatic or semi-automatic acquisition of linguistic knowledge as the preferred route to quickly developing language technology tools for new languages.

2.3 Basic resources and tools

With the foundations in place, the next step is to develop basic processing resources, or language technology tools. Some of these tools are more or less language-independent, and can be reused without modification for new languages, whereas others need language-specific data.

- Statistical tools for corpus treatment (bigram and trigram frequencies, word counts, collocations, etc.)
- Part-of-speech (POS) tagger
- POS-tagged corpora
- Lexical database containing information about parts of speech and morphology
- Morphological analyzer/generator [morphological analyzer]
- Speech recognition systems recognizing isolated words

Among the most basic tools, we find those used for fundamental statistical modeling of linguistic phenomena, where *bigrams* and *trigrams* – sequences of two (*bi-*) or three (*tri-*) units, e.g. letters or words; single units are *unigrams* in this terminology – figure prominently. Another basic tool is the so-called *part-of-speech* (POS) *tagger,* which actually normally (automatically) assigns not only part-of-speech labels in the traditional sense, but full morphosyntactic descriptions, i.e. part of speech and inflectional categories, to all *tokens* (words or punctuation signs) in a text (although they will not assign lemmas, i.e. basic, or citation, forms), in the form of tags, or labels, attached to the words, as in (1), taken from an automatically POS tagged learner corpus (Borin and Prütz 2004), or otherwise refer to the words, as in the interlinear format in (2), from Anju Saxena's corpus of Kinnauri narratives (cf. also the more elaborate SGML format in examples 3 and 4, further below). POS tagging is a special case of what is generally referred to in the context of language technology as (automatic linguistic) *annotation* of text.

(1) Another/DD1 great/JJ fear/NN1 was/VBDZ that/CST wilderness/NN1 would/VM force/VVI civilised/JJ men/NN2 to/TO act/VVI like/II savages/NN2 ./YSTP

(2)   \ref   07/007a/01

    \tx   əma   rəŋ   ᵇoa   loshigyɔ   //

    \mrep   əma   rəŋ   bɔba   lo-sh-i-gyɔ

    \gl   mother   with   father   say-?-?-D.PST

    \tr   Mother and father said:

    \ref   07/007a/02

    \tx   jɔ   tshɛtsats-u   naməŋ   chə   tate   //

    \mrep   jɔ   tshɛtsats-u   naməŋ   chəd   ta-te

    \gl   this   girl-POSS   name(N)   what   keep-LET's

    \tr   "what should we name this girl?

    \ref   07/007a/03

    \tx   naməŋ   tə   sɔthlets   tate   //

    \mrep   naməŋ   tə   sɔthlets   ta-te

    \gl   name(N)   EMP   name   keep-LET's

    \tr   Let's keep the name (=name her) Sothlets."

POS taggers normally assign only one POS label to each word, generally the most probable one for the word form, given the local context (defined as the one, two, or three immediately preceding words with their POS tags). A POS tagger is normally designed so that it will assign a tag to unknown words as well, i.e. words not in its lexicon – also called *out of vocabulary* (OOV) words. Another alternative for achieving almost the same result is to use a *morphological analyzer*, which normally does not take context into account, and consequently assigns all possible analyses (including lemmas) to ambiguous words. Morphological analyzers will not attempt to guess analyses for OOV words, however, but rather mark them as such.

Currently, there are very good POS taggers for English and other similar languages, which are *trained*, i.e. they work with algorithms which are capable of 'learning' from correctly tagged corpora the linguistic knowledge needed for tagging new, previously unseen text (see section 3, below). This of course means that there must be some manually annotated corpus resources available, the larger and the more varied, the better. The alternative, as I have mentioned already, is to use a morphological analyzer with hand-written rules, possibly also together with disambiguation rules for handling the ambiguous cases. In this case there is in principle no need for a pre-annotated corpus, but in practice it will be needed anyway, for automatic objective evaluation of the morphological analyzer. We will return to the issue of automatic acquisition of linguistic knowledge by the computer vs. hand-crafted rule systems in section 3 below.

2.4 Medium-complexity resources and tools

The resources and tools found under this heading all presuppose widespread computer literacy in the language community as well as a noticeable web presence for the language.

- Environment for (available) tool integration – using a standard for representation of linguistic knowledge: XML/SGML, etc.
- Spelling checker and corrector
- Structured lexical databases including multiword lexical units
- Surface syntax analyzer ("chunker") recognizing simple (nonrecursive) constituents and phrases (NP, PP, verb)
- Web crawler managing language X

The *eXtensible Markup Language* (XML) and the older *Standardized Generalized Markup Language* (SGML) are both structured (and very bulky) formats for representing structured documents electronically. They are widely used for representing linguistic resources such as corpora, lexicons, grammars, etc., in an platform-independent way. See examples (3) and (4), which show two different versions of the same sentence from the one-million word balanced Stockholm Umeå corpus of written Swedish (SUC; Ejerhed and Källgren 1997), coded in SGML.

(3) <s id=kl01-001>
<d n=1>-<ana><ps>MID<b>-</d>
<w n=2>Vilka<ana><ps>HD<m>UTR/NEU PLU
IND<b>vilken</w>
<w n=3>djävla<ana><ps>JJ<m>POS UTR/NEU SIN/PLU
IND/DEF NOM<b>djävla</w>
<w n=4>optimister<ana><ps>NN<m>UTR PLU IND
NOM<b>optimist</w>
<d n=5>,<ana><ps>MID<b>,</d>
<w n=6>frustade<ana><ps>VB<m>PRT
AKT<b>frusta</w>
<name type=person>
<w n=7>Lasse<ana><ps>PM<m>NOM<b>Lasse</w>
</name>
<d n=8>.<ana><ps>MAD<b>.</d>
</s>

(4)   <s id=kl01-001>
       <c lem='-' msd='FI' n=1>-</c>
       <w lem='vilken' msd='DH@0P@S' n=2>Vilka</w>
       <w lem='djävla' msd='AQP00N0S' n=3>djävla</w>
       <w lem='optimist' msd='NCUPN@IS' n=4>optimister</w>
       <c lem=',' msd='FI' n=5>,</c>
       <w lem='frusta' msd='V@IIAS' n=6>frustade</w>
       <name type=person>
       <w lem='Lasse' msd='NP00N@0S' n=7>Lasse</w>
       </name>
       <c lem='.' msd='FE' n=8>.</c>
       </s>

The *web crawler* (also: *web spider, web robot*) is one of the corner-stones of current web technology, a computer program which accesses web pages by traversing documents and the links between them and which does something to the pages based upon what it finds; web search engines use web crawlers for indexing web pages. Web crawlers need to be able to apply basic language technology such as token-izers and language recognizers to the text in the web pages. For many languages, the simple forms of indexing used for English may not be sufficient, but some form of lemmatization and, e.g., decompounding may be necessary in order to build efficient information retrieval applications for the language.

2.5   Advanced resources and tools

This is where the bulk of mainstream language technology resource and tool development is being pursued in many language communities, including the smaller Western European languages, such as Finnish, Swedish, Norwegian, Portuguese, etc., some East and Central European languages, such as Russian, Hungarian, Czech, etc., and some South Asian languages, e.g., Hindi.

- Syntactically annotated corpora ("treebanks")
- Lemmatizer[10]
- Grammar and style checkers
- Integration of dictionaries in text editors

---

[10] Sarasola places this technology under the heading of basic tools, together with the morphological analyzer/generator, but I have moved it here to reflect the fact that a good-coverage lemmatizer places much greater demands on the lexical information required than a POS tagger or chunker. A lemmatizer performs morphological ana-lysis together with a disambiguation step – often implemented using a POS tagger – wherein all analyses but one are removed in the case of ambiguous word forms.

- Lexical-semantic knowledge base. Concept taxonomy, e.g. WordNet
- Word sense disambiguation
- Speech processing at sentence level

Treebanks – syntactically annotated corpora – are among the hottest topics in language technology today. There are innumerable treebank projects being pursued all over the world, and quite a lot of research on ways of making the treebanking more labor- and resource-effective; currently it seems that there is a constant labor cost of about one person-year per 50 000 words of (manually corrected) treebank, almost regardless of the language.[11]

Word sense disambiguation (WSD) is the process of (automatically) selecting the correct sense of a polysemous word in context, e.g. *lock* (in canal or in door). WSD is also a research topic of great interest, even to the extent that there are word sense disambiguation competitions – called SENSEVAL – for several languages, not just English.

2.6   Multilinguality and general applications

With this category of resources and tools, we have left the single-language scenario. Here, we are dealing with language technology for multilingual societies, where many languages are used in various administrative capacities. Multilingual language technology in this sense has long been an important item on the research agenda in the European Union, which is founded on the principle that all the official EU languages have equal status within the union, and furthermore that the so-called lesser used languages of the union also receive considerable support. Language technology, being a field dominated by US researchers, has been a bit slower to adopt this view,[12] but there the intelligence community and the military have pressed for language technology support for multilingual information-processing capacity of a slightly different sort. An analogous situation to that of the EU occurs in several places in South Asia, notably in India, where there is now a growing language technology research community.

The "general applications" referred to in the heading demand that we move away from form-centered sentence processing toward, on the

---

[11] This figure is a ballpark estimate based on personal communications from Eckhard Bick (for the Danish treebank) and Jan Hajič (for the Prague dependency treebank), and on published figures pertaining to the work on the ICE-GB parsed corpus of British English (Nelson, Wallis and Aarts 2002).

[12] A telling remark in this context is the following, quoted by Phillipson (2003): "The most serious problem for the European Union is that it has so many languages, this preventing real integration and development of the Union. *The ambassador of the USA to Denmark, Mr Elton, 1997* [endnote I.1:] This remark was made at an informal lunch at the University of Roskilde, Denmark." (Phillipson 2003: 1, 208)

one hand, meaning (linguistic content, semantics) and beyond (world knowledge, encyclopedic knowledge), and, on the other hand, text-level phenomena (discourse and dialogue).

On this level, we find mainly large Western European languages: English, but also French, German, Italian and Spanish. Outside Western Europe and North America, it is mainly Japanese and Chinese which boast at least some of these resources.

- Semantically annotated corpora
- Information retrieval and extraction
- Machine translation systems; translation of NPs and simple sentences
- Dialog systems
- Multilingual lexical-semantic knowledge base
- Language learning systems using human language technology

Predictably, *information retrieval* and *information extraction* play important roles here. Information retrieval is what a web search engine such as Google does; given some keywords, it retrieves all documents (web pages) for which the keywords are relevant (in some sense). Information extraction looks for predetermined pieces of information in documents, e.g. which company acquired which other company, etc. Like WSD, there are competitions *cum* conferences both in information retrieval – the *Text Retrieval Conference* (TREC) – and in information extraction – the *Message Understanding Conference* (MUC). Just like the SENSEVAL WSD competition, other languages than English are represented at these events, as is information retrieval and extraction directly from digitized speech, i.e. spoken language.

Especially information extraction requires some kind of linguistic processing of the documents, but arguably information retrieval does as well, for languages with more complex morphology than English. There is also great interest in so-called cross-lingual information retrieval, where the queries/ keywords are submitted in another language than that of the retrieved documents. This makes perfect sense if you consider that receptive second and foreign language skills are always stronger than the corresponding productive skills, at least in the case of reading vs. writing; you may not be able to formulate the query, but you may well be able to understand the documents, once retrieved.

2.7   The language technology "resource pyramid"

Most of the language technology resources listed in the preceding sections are typically not available for more than a few languages – very few if we reckon with all the approximately 6–7000 languages in

the world – but still few even if we count only those languages having a written form using a standard orthography (see the discussion about basic literacy in section 2.1 above).

There is an important 'refinement' (or 'linguistic ascent') relationship among the resources (or groups of resources). For instance, in the text corpus case, going from a 'mere' collection of texts to a proper corpus entails a kind of selection process, where the main criterion is one of suitability for some particular purpose. The linguistic ascent aspect of the refinement manifests itself as successively more linguistically sophisticated annotations on the corpus (or texts). When annotation is (at least in part) automated, this is normally 'ascent' in a more concrete sense, as well, since annotations on a lower level form the basis for those on a higher level, so that e.g. POS annotation of a text forms the basis for partial parsing of that text. Figure 1 illustrates this dependence among the resources discussed above in a more graphic manner. In the main resource pyramid, there are smaller pyramids, delimited by the slanted lines. The full pyramid is available for languages of type Lg 1 (possibly only English), and many, many languages in the world are of the type Lg 6, having no language technology resources whatsoever. A fair number of languages are found in between, however, and of the four South Asian languages surveyed in more depth as part of the LDC LoDL survey (Bengali, Hindi, Panjabi, and Tamil), Bengali and Panjabi seem to be equipped with the foundations, i.e. they belong to type Lg 5 or Lg 4, whereas for Hindi and Tamil, there are more resources and more sophisticated resources, somewhere between types Lg 4 and Lg 3.

With automated annotation, there is a reciprocal relationship between annotated corpora and the tools used to annotate them. The relationship is one of machine learning. Typically, a pre-existing annotated corpus is used to train an automatic annotator (part-of-speech tagger, parser, etc.), which can then in turn be used to annotate other, previously unannotated corpora, or simply used in some language technology application. This is discussed in considerably more detail in section 3 below.

However, as we have noted above, the initial annotated corpus does not appear out of thin air. Most often, it is hand-annotated by (teams of) human linguists. This is a very time-consuming and labor-intensive effort. As an example, we can mention the Swedish SUC (Stockholm Umeå Corpus), a morphosyntactically annotated and lemmatized one-million-word balanced corpus of modern published written Swedish. It took six years to compile SUC from scratch, and even then, there were still errors in the annotation in the first version, which have been corrected in the second version, which took another three years to complete (Britt Hartmann, p.c.).

Thus, both corpus compilation and above all corpus annotation seem to be very labor-intensive activities. The latter also requires a high degree of linguistic training in the annotators, as well as a general agreement on what a linguistic description of the language should look like, i.e. an agreed-upon tradition as regards terminology, etc. It seems reasonable to assume that lesser-known language communities – especially those where language standardization is recent or in progress – will have few trained linguists and possibly no descriptive linguistic tradition to draw upon.

Even in the case of a well-described major language, however, the cost of annotation may be prohibitive – it is no coincidence that we find, for a number of languages, that there are at least some resources available belonging to the lower levels on the resource pyramid – there are many unannotated corpora and a fair number of POS-tagged corpora for many languages – but e.g. that even for English the number of treebanks can be counted on the fingers of one hand.[13] Hence, there is a fair amount of research in the language technology community addressing the issue of how to minimize the human effort in corpus annotation.[14]

---

[13] And now, the web can be used as a source of (a kind of) corpora, for assembling them on the fly, as it were (Ghani, Jones and Mladenić 2001a, 2001b; De Schryver 2002; Nilsson and Borin 2002; Maynard, Tablan and Cunningham 2003; Nilsson 2003; Oard et al. 2003). Of course, the languages must be written languages, and there must be a sufficient number of web publications in them. Of the two, the first is the more restrictive requirement, since only a modest fraction of the world's languages are written. Even written languages are quite unevenly represented on the web, however; in another connection, I endeavored to survey the availability on the web of material in two official minority languages in the Nordic area, Sámi and Finnish Romani. It turned out that while Sámi (mainly North Sámi) had a small web presence, Finnish Romani was, for all practical purposes, not represented at all on the web (Nilsson and Borin 2002: 416–417).

[14] Note that this actually means minimizing human effort in the development of all sorts of language technology applications, since many such applications can be seen as special cases of corpus annotation.
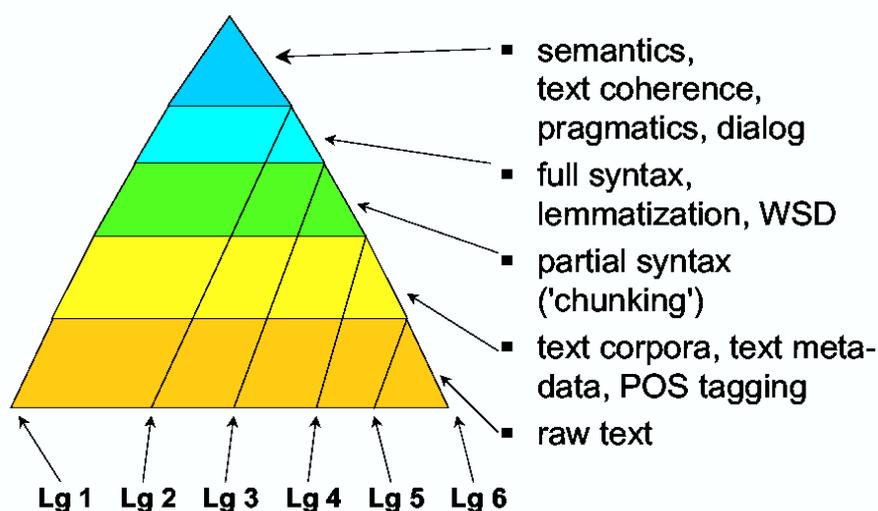
Figure 1:   *The language technology resource pyramid*

3   The solution: Bootstrapping

A procedure where linguistic annotation of corpora is accomplished by starting out with a small amount of hand-annotated material (or none at all), and by (partly) automated means developing increasingly more sophisticated and correct automatic annotators from this basis, is known in the literature as "bootstrapping",[15] and this is the term I will use here. In the following, I will attempt to give an overview of the current state of the art in this research area, which has seen noticeably intensified activity in the last few years.

3.1   Machine learning of linguistic knowledge

An essential and defining component of bootstrapping is *machine learning*, a subdiscipline of computer science (or of artificial intelligence) which concerns itself with self-learning computer systems. Rather than programming a computer with explicit rules for performing some task, machine learning algorithms are supposed to enable the computer to learn the regularities underlying the explicit rules, and, of course, to apply the knowledge thus acquired, in the same way as an explicitly programmed system would do it. Most –

---

[15] The metaphor behind the term comes from the the English expression *to pull one-self up by one's own bootstraps*: "The term *bootstrapping* here refers to a problem setting in which one is given a small set of labeled data and a large set of unlabeled data, and the task is to induce a classifier. The plenitude of unlabeled natural language data, and the paucity of labeled data, have made bootstrapping a topic of interest in computational linguistics." (Abney 2002: 360) Here, we will assume the term to also include the setting where no labelled data are available, i.e., pure inductive (or abductive) learning.

perhaps all – learning tasks in this area are logically reducible to the problem of *classification*, i.e. the computer learns to classify items, or instances, as belonging to particular types, or categories. For this reason, machine learning algorithms are often referred to as *classifiers* in the literature. It will come as no surprise that self-learning systems have become most popular for classification problems where no good rule formats seem to exist, or where the relevant rules are unknown, e.g. for face and voice recognition, for detecting credit card fraud – i.e., for classifying credit card transactions as normal or as somehow deviant – etc. For the purpose of classifying, or labelling, linguistic units in texts, there are at least seven kinds of linguistic units that have been subject to machine learning research:

1. words (i.e., division into words;[16] level 0 in "the resource pyramid"; see section 2, above);
2. parts of speech, or morphosyntactic categories in general (level 2);
3. morphological regularities (i.e., learning morphology; level 2);
4. syntax (level 3–4);
5. (lexical) semantics, including word sense disambiguation and named entity (NE) recognition (level 4–5);
6. text-level phenomena, such as dialogue acts and coreference (level 5–); and
7. translation equivalents (level 1).

Machine learning is a large and varied discipline (see Mitchell 1997 for a general textbook level introduction to the field, and Manning and Schütze 1999 for a more language technology-oriented treatment).

3.2   Kinds of machine learning

In machine learning, there is a fairly sharp dividing line between systems which are able in some way to "justify" their classifications, and those that cannot do this in a meaningful way. Note that there is in general no requirement that a classification should be justified in some way in order for it to be useful. The prototypical case here is that of a parent saying to a child: "Because I say so!". This dividing line largely coincides with that between *statistical* (or *stochastic*) and *symbolic* (or *categorical*) machine learning methods, with reference to the character of the regularities resulting from machine learning, namely whether they directly utilize probabilities or not.[17] The symbolic methods have the advantage over the statistical ones – at

---

[16] A division into words is necessary at least in two separate cases: (1) in dealing with orthographies which do not regularly mark word boundaries, the best-known example being Chinese; (2) in segmenting continuously transcribed speech, e.g., certain kinds of automatic transcription.

least potentially – that "the acquired knowledge is represented in a form that is more easily interpreted by human developers and more similar to representations used in manually developed systems. Such interpretable knowledge allows for greater scientific insight into linguistic phenomena, improvement of learned knowledge through human editing, and easier integration with manually developed systems" (Mooney 2003: 377).

Further, machine learning methods can be *supervised* or *unsupervised*. In work on language technology, unsupervised generally means that the learning takes place on the basis of unlabelled data, although the label set itself may be known (this latter use of the term differs from that in the general machine learning literature, where "unsupervised" refers to training on unlabelled data with no given prior label set). To clarify: The terms "labelled" and "unlabelled" here refer to the labels, or classificatory categories, to be learned by the machine learning method. Thus, "unsupervised" in the prototypical machine learning usage of the word refers to pure induction; the computer infers not only the particular classification of the training data, but also the categories themselves, in terms of which this classification is made. The data (the corpus) can be labelled with labels in some other, 'lower', linguistic domain, e.g. an unsupervised phrase structure syntax learner may work with data labelled for part of speech. On closer scrutiny, it turns out that we are not dealing with a simple dichotomy of supervised – unsupervised, but rather that there is a continuum from 'fully supervised' to 'fully unsupervised' learning of linguistic regularities, with many gradations in between the two extremes. For simplicity, however, I will still present this dimension as a dichotomy here, using "unsupervised" to mean "not fully supervised", thus placing all use of *a priori* knowledge other than a fully explicit and exhaustive target labelling in the 'unsupervised' category.

From a linguist's point of view, however, we still might want to discern a kind of 'semi-unsupervised' machine learning regime as well, where some probabilities are fixed *a priori*, on the basis of knowledge of language universals and linguistic typology. "Semi-unsuper-vised", rather than "semi-supervised", since – although rarely encountered in the language technology literature – this regime is mainly used in otherwise unsupervised machine learning, adding an element of abduction, as it were, to an otherwise inductive method. It is rare enough that it does not make sense to make it a part of the overall typology; rather, I will return to this issue in the next section.

---

[17] It is a different issue altogether as to whether the machine learning method used involves probabilities. Arguably, all such methods do, at least implicitly.

3.3   Two linguistic examples: Statistical and symbolic POS tagging

In order to hopefully make things clearer to the reader unacquainted with language technology, I will here briefly describe two forms of supervised machine learning that are among the standard tools of modern language technology. The linguistic domain used to illustrate how the methods work will in both cases be POS tagging, although both methods are more general, and have also been used for learning in other linguistic domains. I will describe one statistical and one symbolic method.

3.3.1 POS annotation with statistical n-gram tagging

A widely used method for POS tagging of text is the statistical *Hidden Markov Model* (HMM), or *n-gram*, tagger. This tagging technology works exclusively with probabilities, approximated by relative frequencies aquired from a training corpus, plus some probability set aside for unseen words and POS tags. Basically, a n-gram tagger uses two kinds of probabilities.

First, there are the so-called *lexical probabilities*, i.e. the probability of a particular token having a particular POS tag, regardless of its context. These are also called *unigram* probabilities. In the training corpus, we may find that a particular token appears 872 times as a noun, and 128 times as a verb. Then the *a priori* lexical probabilities will be close to 0.872 [872 / (872+128)][18] for the noun tag, given the token, and thus close to 0.128 for the verb tag. The actual probabilities used will usually be a bit lower (hence the phrasing "close to"), since we may want to assign some small probability to other (open class) tags as well, on the assumption that they might just accidentally not have shown up in our particular training corpus (and the probabilities should sum to unity). For English, using only the lexical probabilities (consistently assigning the highest-probability tag) will typically yield an accuracy (i.e., percentage correct tags) of around 90%. No tagger should ever do worse than this baseline, that is.

The other kind of probabilities are the so-called *contextual probabilities*. Basically, they boil down to the probability of a particular POS tag appearing in the context of the *n* preceding POS tags in the same sentence. Normally, *n* is 1 or 2 (rarely larger), yielding *bigram* or *trigram* models (including the current POS tag). These probabilities are calculated from the actually occurring POS tag n-grams in the training corpus, again with some probability saved for (accidentally) non-occurring sequences.

---

[18] That is, the number of noun occurrences of the word divided by its total number of occurrences.

From the two kinds of *a priori* probabilities, some mathematical magic is then used to derive the set of probabilities which will best "explain" the POS tag sequences of the corpus, given the token sequences. These probabilities are in turn then used to annotate unseen texts with POS tags, achieving a typical accuracy (for English) on the order of 95–98%, depending on the tagset used and on how similar the unseen text and the training text are. For a more detailed explanation of HMM tagging and related statistically based methods, see Manning and Schütze 1999.

3.3.2 POS annotation with transformation-based learning

Now we turn to the use of a symbolic machine learning method for the same kind of linguistic analysis task, i.e. POS tagging. The method is more generally known as *transformation-based learning* (TBL), but its use for POS tagging first became popular under the – still often used – name of *Brill tagging*, after its inventor, Eric Brill (Brill 1993, 1995). Transformation-based learning is a supervised machine learning method, just like the HMM tagger described above. However, instead of probabilities of various kinds, a TBL system learns *transformation rules*. For POS tagging, the rules have the general format "if the POS tag of the current token is X and the stated contextual and other conditions are true, change the POS tag of the current token to Y". This of course implies that the corpus should have an initial annotation, which is accomplished by tagging tokens with their most probable POS tag according to the training corpus (i.e., the highest unigram probabilities are used; see above). OOV tokens are given some high-probability tag; in the case of English, this will normally be common noun or proper noun, the latter only in the case of capitalized, sentence-internal words.

The TBL system will also be equipped with a general template for transformation rules, especially with respect to possible contextual and other conditions. Typically these may refer to tokens and tags, up to, say, two positions on either side of the current token. Additionally, it often makes sense to be able to refer to token endings or beginnings. Thus, a typical rule for English may be paraphrased as "if the POS tag of the current token is V(erb), and the POS tag of the immediately preceding token is Det(erminer), then change the POS tag of the current token to N(oun)". The TBL system will start by generating all possible such rules, licensed by the templates and the actually occuring token sequences in the training corpus. It will then start evaluating rules according to how many errors each rule removes, and how many errors it introduces, when applied to the entire training corpus. Rules will be ordered according to this, and those falling below a certain threshold will be removed altogether.

Ordering is important; just like the more familiar tranformation rules of generative linguistics, these rules, too, may feed or bleed each other. Thus, the TBL system normally iterates a number of times, reordering and removing rules, until no (significant) changes take place on two consecutive runs through the whole training corpus. Applying the resulting tagger on unseen text then follows the same procedure, apart from the rule evaluation. First, the unseen text is annotated using the most likely POS tag for each token, and then the transformation rules are applied in order. Brill taggers tend to achieve an accuracy that is comparable to that of statistical n-gram taggers, but do it in such a way that it "captures linguistic information in a small number of simple nonstochastic rules, as opposed to large numbers of lexical and contextual probabilities" (Brill 1995: 563).[19]

3.4   Bootstrapping methodologies and language technology

From the above, it is natural to discern four machine learning, or bootstrapping, 'methodologies':

(1)   supervised statistical;
(2)   unsupervised statistical;
(3)   supervised symbolic; and
(4)   unsupervised symbolic.

In Table 1, I provide pointers to recent literature on bootstrapping of linguistic annotations in terms of these distinctions, "recent" meaning here roughly "published later than 1990". Those interested in earlier work on machine learning of morphology in particular may consult Borin 1991, especially chapter 6.

Not all possible combinations occur in Table 1, of course. Also, in some cases (e.g., statistical part of speech taggers and chunkers, Brill taggers, probabilistic context-free parsers), the technology is so mature and well-known in the LT community that I have deemed it unnecessary to cite specific references in the table. For statistical POS taggers and Brill taggers, the reader is referred to the brief explanations provided above, and for more details on those and also on chunkers and probabilistic context-free parsers, to general reference works and textbooks such as Cole et al. 1996, Manning and Schütze 1999, Jurafsky and Martin 2000, or Mitkov 2003.

---

[19] Although it has to be said that the number of lexical probabilities will be the same in both approaches.

Table 1: Overview of works on bootstrapping in language technology

ML METHODOLOGY	LINGUISTIC DOMAIN
STATISTICAL SUPERVISED	
statistical taggers	POS tags
statistical chunkers	base (syntactic) phrases
probabilistic context-free parsers	syntactic (phrase) structure
STATISTICAL UNSUPERVISED	
Elworthy 1994; Merialdo 1994; Clark 2003; Clark et al. 2003	POS tagset
Clark 2001, 2002; Lee et al. 2003	morphology
Keller and Lütz 1997a, 1997b	syntactic (phrase) structure
SYMBOLIC SUPERVISED	
Brill taggers; memory-based tagger; Cucerzan and Yarowsky 2002	POS tags
transformation-based chunkers	base (syntactic) phrases
Ngai and Yarowsky 2000	base NPs
SYMBOLIC UNSUPERVISED	
Yarowsky and Ngai 2001; Yarowsky, Ngai and Wicentowski 2001; Borin 2002b; Clark 2003	POS tags
Rogati, McCarley and Yang 2003	stemming
Yarowsky, Ngai and Wicentowski 2001	named entities (NE)
Yarowsky and Ngai 2001; Yarowsky, Ngai and Wicentowski 2001	base NPs
Merkel 1999; Borin 2000a, 2002a (and references given there); Schafer and Yarowsky 2002; Martin et al. 2003; Tiedemann 2003	translation equivalents
Schone and Jurafsky 2000, 2002; Goldsmith 2001; Snover and Brent 2001, 2002; Yarowsky, Ngai and Wicentowski 2001; Belkin and Goldsmith 2002; Baroni, Matiasek and Trost 2002; Creutz and Lagus 2002; Sharma, Kalita and Das 2002; Snover 2002; Snover, Jarosz and Brent 2002; Creutz 2003; Johnson and Martin 2003	Item-and-Arrangement (IA) morphology (form concatenation)
Borin 1991 (and references given there); Theron and Cloete 1997; Manandhar, Dzeroski and Erjavec 1998; Manning 1998; Yarowsky and Wicentowski 2000; Neuvel and Fulop 2002	Item-and-Process (IP) and paradigmatic morphology (form relationships)

4   Three bootstrapping scenarios (for lesser-known languages?)

In this section, we will have a closer look at three bootstrapping scenarios, both because they are fairly well-researched and because they seem promising for the problem of creating annotated language technology resources for lesser-known languages. At the same time, there are some theoretically interesting questions as to their general applicability, which I will address in section 5.

4.1   Unsupervised or 'lightly supervised' learning of linguistic generalizations from corpora

This is the scenario which would be most useful, were it to be realized even in part. Basically, we are talking about pure inductive (or possibly abductive) learning of linguistic regularities, of the kind envisioned by pre-Chomskyan American structuralists (e.g. in several of the papers reprinted in part 1 of Harris (1970); Garvin (1967); see also Borin 1991, ch. 6)). Table 1 provides the basic references, and here I will briefly review this literature and the methods proposed there, focusing on the problem of learning inflectional morphology directly from an unannotated corpus. This is an interesting and important problem, since many of the languages of the world have more morphology than English – out of the 313 languages in the LDC LoDL survey, only 41 are listed as unequivocally having a "simple morphology" (54 have a non-simple morphology, and for the remainder apparently there were insufficient data). In a language having a 'non-simple' morphology, morphological analysis will presumably be useful or needed for carrying out other annotation tasks, such as lemmatization and syntactic analysis.

   In the literature the problem of learning morphology is sometimes seen as involving only the ability to relate word forms among themselves in a pairwise fashion, without any attempt at segmentation (e.g., Baroni, Matiasek and Trost 2002). In other cases, the aim is for learning quite general string transformation regularities, i.e. morphology learning is viewed in general IP kind of framework (e.g., Theron and Cloete 1997; Clark 2001, 2002; Neuvel and Fulop 2002). However, most works on morphology induction propose to factor out common substrings among the words in the corpus, segmenting word forms into nonoverlapping pieces – thus opting for an IA (or concatenative) model of morphology – most commonly so that words are divided into a stem and a suffix. In other words, a 'Standard European' kind of inflectional morphology is posited (see section 3, below), although we also find attempts to learn recursive (i.e., stem+affix structures, where stems in turn are seen as made up of stem+affix; e.g., in the *Linguistica* morphology learning program

described by Goldsmith 2001) and iterative (i.e., morph(eme) sequences; e.g., Creutz and Lagus 2002; Creutz 2003) structures, as well as prefix-suffix combinations (or circumfixes; e.g., Schone and Jurafsky 2000, 2002).

Various methods have been proposed for deciding which forms should be related to one another, and where to make the cuts in the word forms. In the most commonly used approach, the factorization involves some kind of information theoretic or probability measure, which is used to calculate the overall best division point between stem and suffix, or division points between morphs. Very common here is the use of MDL (Minimum Description Length; Rissanen 1978, 1989), as in the work of Goldsmith (2001) and of Creutz and Lagus (2002). In practice, this means that the problem of morphology induction is seen as being of a kind with that of data compression. The main difference compared to data compression is that normally at least some *a priori* linguistic knowledge is assumed, and allowed to constrain the information theoretic measure in appropriate ways. Thus, existing word boundaries (spaces, etc.) are taken as given (unlike the case when a text file is compressed merely to save space).[20] Instead of MDL, some researchers have posited *a priori* probabilistic models to which morpheme lengths and frequencies should be fitted (e.g., Snover 2002; Snover and Brent 2001, 2002; Snover, Jarosz and Brent 2002; Creutz 2003).

There are also approaches which do not use probability or information-theoretic measures at all, but instead seek purely discrete relatedness measures and symbolic factorizations, e.g. by calculating the minimum edit distance (or Levenshtein distance; Kruskal 1983) between pairs of word forms (e.g. Theron and Cloete 1997; Yarowsky and Wicentowski 2000; Baroni, Matiasek and Trost 2002), or by storing words in a *trie* (see Knuth 1973) which is subsequently analyzed as to its topology (Schone and Jurafsky 2000, 2002), or transformed into a minimal acyclic finite-state automaton (Johnson and Martin 2003; for minimal finite-state automata, see Watson and Daciuk 2003).

All these methods tend to overgenerate, however, and consequently we find proposals in the literature for improving their results, by 'pre-biasing' or 'filtering' (if we had been talking about machine translation, we would have said "pre-editing" or "post-editing") their results. All proposals boil down to using additional sources of information deemed relevant for the morphology learning problem.

Starting with those approaches where 'filtering' is applied to clean up noisy machine learning results, we find heuristics such as elimination of singly occurring 'stems' and 'affixes', i.e. each proposed stem and affix should appear at least twice or it will be removed from con-

---

[20] Spaces are normally eliminated in a preprocessing step, so that the 'text' to be 'compressed' is actually a set of word forms (a word form list).

sideration (e.g., Goldsmith 2001). We also find explicit models of inflectional paradigms being used to group affixes and classify stems (e.g. Goldsmith 2001; Snover 2002; Snover and Brent 2001, 2002; Snover, Jarosz and Brent 2002).

Further, we find attempts to use syntax, in the form of near context, to separate homonymous stems or affixes according to their parts of speech or functions, respectively (e.g. Yarowsky and Wicentowski 2000; Goldsmith and Belkin 2002; Schone and Jurafsky 2002), and attempts to use semantics in the form of mutual information (e.g., Baroni, Matiasek and Trost 2002), or LSA (Latent Semantic Analysis; Landauer, Foltz and Laham 1998), to separate homonymous stems and affixes according to their meanings or functions, respectively, and to eliminate spurious segmentations, such as *all-y*, derived from *all* (e.g., Schone and Jurafsky 2000, 2002).

The use of 'pre-biasing' takes diverse forms. Most common is perhaps a lexical kind of bias, where word forms are grouped into their proper paradigms, and the task of the morphology learner is to associate form relations with (explicitly stated) meaning relations (e.g., Theron and Cloete 1997; Oflazer and Nirenburg 1999; Oflazer, McShane and Nirenburg 2001). Sometimes, an affix inventory (full or partial, unstructured or structured into parts of speech) is provided beforehand (e.g., Cucerzan and Yarowsky 2002; Rogati, McCarley and Yang 2003). Finally, purely probabilistic models can have model parameters fixed in advance (e.g., the most common morph length and *hapax legomena* proportion in the model of Creutz 2003).[21]

Additional information can also be provided using another language or by setting up the machine learning so that it involves interaction with human experts at appropriate points – so-called *active learning*. These two cases are special enough in our context that they deserve separate treatment in the following two subsections.

4.2   Cross-language annotation transfer

Given the common situation of a dominant language which has some language technology resources coexisting in one political entity with a lesser-known language which lacks some or all of these resources, but where there are fair amounts of (machine-readable) parallel texts in the two languages, the idea naturally introduces itself of trying to transfer dominant language annotations into the lesser-known language via an alignment of the parallel texts on some linguistic level (see e.g. Borin 2002b for a discussion of the general idea, although not applied to dominant–lesser-known language pairs).

---

[21] Here, we could imagine including prior information of a more general kind as well in a morphology learner, for instance a preference for suffixes over other kinds of affixes, which is characteristic of the world's languages as a whole.

How well this will work out is dependent on a number of factors, e.g. the kind of annotation targeted and the closeness of the languages involved (see Trosterud 2002), but in some cases it could be used in order to get a first rough annotation which could then be refined using a mix of human correction and automatic methods, as discussed in the next section.

A special case of this methodology would be to use another language indirectly, as it were, using an annotation tool trained on some language A for annotating a different language B. Maynard, Tablan and Cunningham (2003) do exactly this when they apply an English named entity recognizer to Cebuano (an Austronesian language of the Philippines).

Although there has been considerably less research on this problem than on monolingual bootstrapping, researchers have endeavored to transfer at least the following kinds of annotation across languages in this fashion: lemmas (the extensive research on word alignment; see Borin 2002a); part of speech tags (Yarowski and Ngai 2001; Yarowski, Ngai and Wicentowski 2001; Borin 2002b); base NPs (Yarowski and Ngai 2001; Yarowski, Ngai and Wicentowski 2001); morphological analyses (Yarowski, Ngai and Wicentowski 2001); and morphemes (Johnson and Martin 2003).

4.3 Computer-assisted human annotation (ᴏʀ human-assisted computer annotation)

This is the most realistic scenario, representing a more sober assessment of the present capabilities of induction of linguistic regularities by machine learning than the first scenario, stating that its proper role is as an assisting technology for human annotators, rather than as a fully automated process.[22]

Especially promising is the simultaneous use of more than one source of (linguistic) information in concert, thereby achieving a result that is more than the sum of the parts. In this vein, there has been work on combining small amounts of linguistic knowledge (elicited from native speakers or taken from reference works written for humans) with various kinds of machine learning (e.g. Oflazer and Nirenburg 1999; Oflazer, McShane and Nirenburg 2001; Cucerzan and Yarowsky 2002; Neuvel and Fulop 2002), on how to best present language data to machine learning algorithms and how to select the most useful data items for human annotation (Engelson and Dagan

---

[22] This is reminiscent of the development in the field of machine translation (MT). In the beginning, there were high hopes for fully automatic high-quality MT, which were never realized. Only with the 'ideological' reorientation of the field towards machine-aided human translation as the focal application area has MT started enjoying some kind of commercial success.

1996; Abney 2002; Steedman et al. 2003), and on the most cost-effective combination of human and machine annotation (Ngai and Yarowsky 2000).

5   English-only world?[23]

It could be argued that language technology has been shaped by the typological and other traits of the most explored language, namely English. These traits are, i.a.

1. inflectional morphology with very few forms (two main and two marginal noun forms, four verb forms, uninflected adjectives, except for comparison forms in a few cases) [⇒keeps type-token ratio[24] down];
2. not much in the way of derivational morphology [⇒keeps type-token ratio down];
3. weak formal separation of parts of speech;
4. fairly rigid word order [⇒works well with simple phrase structure formalisms];
5. etymological spelling keeping homophonous items separate in writing [⇒keeps (semantic) types separate];
6. word separation markers in the orthography [⇒keeps type-token ratio down];
7. orthographic marking (capitalization) of proper nouns [⇒keeps (semantic) types separate];
8. compound parts written as separate words [⇒keeps type-token ratio down].
9. little non-concatenative morphology

However, English is in some respects an atypical language, and it would consequently be a mistake to believe that traits such as the ones listed and others will be characteristic of all or a large number of languages. I hasten to add that these traits are found in other languages, too, and not only in those genetically or geographically close to English. Thus, Chinese shares at least traits 1–5 with English, while 6–7 work differently (no word boundary markers and no special indication of proper nouns). My point – which I am not the first to have made – is simply that there is an abundance of languages which work differently from English, and the question then rightly raises

---

[23] See Phillipson 2003.

[24] The *type-token* ratio of a text is calculated by dividing the number of *different* words (and sometimes punctuation signs) – the *types* – by the total number of words (and sometimes punctuation signs) – the *tokens*. This is a measure (although indirect) of the vocabulary diversity of the text, and its inverse gives the sample mean, i.e., the average number of occurrences of a text word. Importantly, the type-token ratio is not constant, but decreases nonlinearly with text length. See Baayen 2001.

itself as to whether the same language technology methods which have worked so well for English will work equally well for languages drastically different from English in these and other respects.[25]

In the way of a small illustration of this, I have applied Goldsmith's (2001) *Linguistica* program to a text in Greenlandic, with the following result (" + " marks the morphological divisions of text word forms into 'stem' and 'suffix', as inferred by the program):

> imaqarnersiuineq ulloq 1 7 . december 2 0 0 2 danmark usa-miit
> kissaateqarfigineqarpoq pituffimmi radari + p
> nutarteriviginissaan + ut akuerineqarni + ssaq pillugu ,
> taamaasilluni taanna missiili + nut illersornissa + mi
> ilaatinneqarsinnaanngorlugu . pingaartuuvoq missiili + nut
> illersuum + mut tunngati + llugu apeqquti + t tamarmik
> sukumii + sumik isumalioqqutigeqqaarnissaat qallunaa + t
> naalakkersuisui + niit amerikamiut kissaat + aat
> akineqalersinnagu . taamaattu + mik naalakkersuisu + t
> kissaatigaat qaammatini aggersuni missiili + nut illersuuti + t
> pillugit oqallin + nerit ingerlateqqinneranni peqataa + nissaq .
> oqallin + nissamut tamatumunnga pilliuti + tut missiili + nut
> illersuuti + t pillugit nassuiaasia + q manna naalakkersuisu + nit
> suliarineqar + poq . nassuiaasia + mi missiili + nut illersuu + t
> pillugu apeqquterpassuit pissutissali + mmik
> saqqummiunneqarsinnaa + sut erseqqissaaviginiarneqarput .
> naalagaaffi + it arlallit missiili + nik ballistiskiusu + nik
> ungasissu + mut anngussinnaasu + nik
> piorsaaviginninnissa + mik aamma + lu sorpassuar + nik
> aseruisinnaasu + tut sakkussia + nik , tamatuma + ni aamma
> atomi + mik sakkussia + nik , pissarsinissa + q pimoorullugit
> sulissutigaat .

We see that the text is segmented to some extent, but still a far cry from what we would expect, knowing even a little about Greenlandic. Goldsmith's program assumes an English-like – or perhaps "Standard Average European" – inflectional system, where words typically con-

---

[25] It could be argued that many of the researchers working in the field of machine learning of linguistic regularities are either English speakers or computer scientists, or both. In both cases there may be an lack of awareness –a benevolent interpretation – or disregard – seeing matters a bit more cynically – on the part of these researchers of such matters as language diversity, language typology, etc. On the other hand, it is often difficult to get linguists to take in, let alone voice an opinion on heavily mathematical work on statistical machine learning. Let it be said immediately that neither holds for of Goldsmith, whose work I will have occasion to refer to below. Sparck Jones puts it very aptly when she says: "It has also to be recognized that the arrogance so characteristic of those connected with IT – the self-defined rulers of the modern world – is not merely irritating in itself, it is thoroughly offensive when joined to ignorance not only of language, but of relevant linguists' work" (1996: 13), and: "On the practical side, it is impossible not to conclude that many linguists are techno- and logico-phobes." (1996: 13f).

sist of a stem – possibly containing weakly productive derivational morphemes – followed by a single inflectional suffix – which, however, sometimes encodes several functions simultaneously – (trait 1 in the list above), but Greenlandic is instead of a linguistic type that is sometimes called "polysynthetic", where a large number of highly productive derivational and inflectional morphemes – suffixes in the case of Greenlandic – are added to root morphemes – sometimes accompanied by morphophonological changes in the morphemes – creating very long words. These derivational and inflectional morphemes in many cases appear instead of the so-called function words in a language like English.[26]

Thus it does not come as a surprise that the type-token ratio for this text is around 0.43 (3480 types / 8084 tokens), which can be compared to another language (chosen for convenience) – namely Finnish Romani – where 8048 word tokens correspond to 1147 word types, yielding a type-token ratio of approximately 0.14. English would presumably show an even lower figure.[27] This is relevant, because many of the fully or partly automatic methods proposed for the bootstrapping scenarios outlined above rely directly or indirectly on probabilities, or statistics, to do their work, and the basic unit that they work with is the orthographic word. In English, the orthographic word is very close in size to the basic lexical unit in linguistic descriptions of English, and it is arguably a great help to these automatic methods that English comes "pre-digested", as it were, i.e. pre-segmented into orthographic words of roughly the right size. The more instances of a word type the automatic methods can work with, the more certain their predictions about the behavior of that type will become, and conversely, if the type frequency falls below some threshold, they will be unable to say anything about it, basically.[28]

---

[26] Goldsmith – being a well-known linguist gone computational – is of course well aware of all these facts, and the first version of *Linguistica* was expressly designed to handle languages of the inflectional type. He has, however, subsequently developed *Linguistica* further, so that the program should now be able to deal with affix sequences as well. There is a general point to be made here, however, which is that the magnitude of the problem grows quickly into the intractable if we cannot assume a fixed and fairly low number of morphs in the word. The prepackaging of English – partly because of morphological type, partly because of orthography – really saves an incredible amount of computational work.

[27] With respect to the linguistic characteristics most relevant for our purposes here, Finnish Romani – a modern Indo-Aryan language – is more like Russian or German than English, but still much closer to English than to Greenlandic (see e.g. Vuorela and Borin 1998; Borin 2000b). Cf. the figures given by Creutz (2003) for his 200 000 word English and Finnish corpora, with 17 000 and 58 000 word types, respectively, yielding type-token ratios of 0.085 for English, but 0.29 for Finnish.

[28] Goldsmith's program is indirectly affected by this, too, since it relies on recurring 'stems' and recurring 'suffixes' in order for it to do its job properly – in the same way that a POS tagger relies on recurring word forms and word form sequences – and the morphological characteristics of a language such as Greenlandic ensure that most 'stems' (in this sense) will be unique and, for the same reason, that most word

As I have tried to indicate above, several traits of English conspire to make current probabilistic models work well even with quite small amounts of English text, by jointly "keeping down" the type-token ratio of English, as compared to many other (written) languages. On the other hand, English might lose some of its advantage if corpora were to come segmented into morphs instead of words, since derivational relationships in other languages tend to correspond to lexical relationships in English (e.g., an English noun corresponding to a Latinate adjective, as in *cat – feline*, etc.): "The proportion of unrecognizable morphemes [in the Finnish test data] is highest for the smallest corpus size (32.5%) and decreases to 8.8% for the largest corpus size" and "the proportion of unseen morphemes [in the English test data] that are impossible to recognize is higher for English (44.5% at 2000 words, 19.0% at 200 000 words)" (Creutz 2003 n.p.).

6   Wishlist for the future

A research program which follows quite naturally from the above would look roughly like this: Begin (systematic) testing of the methods proposed for rapid resource collection and annotation – e.g. those mentioned in section 4 above –, with some "non-English" language as the target language. In the South Asian area there are many good "non-English" candidate languages, and the choice here obviously will depend on which particular linguistic traits are considered most incompatible with the methods in question. If morphological complexity is considered important, as arguably is the case with the various morphology induction algorithms proposed in the literature, then probably some Dravidian language will make up the most appropriate testing ground.

Further, it would be worth exploring annotation transfer via *comparable corpora* – a virtually uncharted territory – which would make much sense to attempt, since comparable corpora are generally easier to come by than parallel corpora (see Borin 2002a for the terminology).

The purpose of such exercises would be first and foremost purely scientific: We would like to get a better understanding of the generality or language-specificity of these methods. At the same time, we might conceivably get the embryo of some language technology resources for the language in question, at least on the first two levels (the *foundations* and *basic resources and tools*) in the overview given above.

---

forms will be unique.

References

Abney, Steven. 2002. "Bootstrapping". *Proceedings of the 40th annual meeting of the ACL*, 360–367. Philadelphia: ACL.

Baayen, R. Harald. 2001. *Word frequency distributions*. Dordrecht: Kluwer.

Baroni, Marco; Matiasek, Johannes; and Trost, Harald. 2002. "Unsupervised discovery of morphologically related words based on orthographic and semantic similarity". *Morphological and phonological learning: Proceedings of the 6th workshop of the ACL special interest group in computational phonology (SIGPHON)*, 48–57. Philadelphia: Association for Computational Linguistics.

Belkin, Mikhail; and Goldsmith, John. 2002. "Using eigenvectors of the bigram graph to infer morpheme identity". *Morphological and phonological learning: Proceedings of the 6th workshop of the ACL special interest group in computational phonology (SIGPHON)*, 41–47. Philadelphia: ACL.

Borin, Lars. 1991. The automatic induction of morphological regularities. Reports from Uppsala University Linguistics (RUUL) #22. Dept. of Linguistics, Uppsala University.

Borin, Lars. 2000a. "You'll take the high road and I'll take the low road: Using a third language to improve bilingual word alignment". *Proceedings of the 18th International Conference on Computational Linguistics, Vol. 1*, 97–103. Saarbrücken: Universität des Saarlandes.

Borin, Lars. 2000b. "A corpus of written Finnish Romani texts". *LREC 2000. Second international conference on language resources and evaluation. Workshop proceedings. Developing language resources for minority languages: Reusability and strategic priorities*, 75–82. Athens: ELRA.

Borin, Lars. 2002a. "… and never the twain shall meet?" In: Borin, Lars (ed.), *Parallel corpora, parallel worlds*, 1–43. Amsterdam: Rodopi.

Borin, Lars. 2002b. "Alignment and tagging". In: Borin, Lars (ed.), *Parallel corpora, parallel worlds*, 207–218. Amsterdam: Rodopi.

Borin, Lars. 2006. "Supporting lesser-known languages: The promise of language technology". In: Saxena, Anju; and Borin, Lars (eds), *Lesser-known languages of South Asia: Status and policies, case studies and applications of language technology*, 317–337. Berlin: Mouton de Gruyter.

Borin, Lars. 2009. Linguistic diversity in the information society. *Proceedings of the SALTMIL 2009 workshop on Information Retrieval and Information Extraction for Less Resourced Languages*, 1–7. Donostia: University of the Basque Country.

Borin, Lars; and Prütz, Klas. 2004. "New wine in old skins? A corpus investigation of L1 syntactic transfer in learner language". In: Aston, Guy; Bernardini, Silvia; and Stewart, Dominic (eds), *Corpora and Language Learners*, 69–89. Amsterdam: John Benjamins.

Brill, Eric. 1993. A corpus-based approach to language learning. University of Pennsylvania Dept. of Computer and Information Science Ph.D. dissertation.

Brill, Eric. 1995. "Transformation-based error-driven learning and natural language processing: A case study in part of speech tagging". *Computational Linguistics* 21(4): 543–565.

Clark, Alexander. 2001. "Learning morphology with pair hidden Markov models". *Proceedings of the student workshop at ACL 2001*. Toulouse: ACL.

Clark, Alexander. 2002. "Memory-based learning of morphology with stochastic transducers". *Proceedings of ACL 2002*. Philadelphia: ACL.

Clark, Alexander. 2003. "Combining distributional and morphological information for part of speech induction". *Proceedings of EACL 2003*.

Clark, Stephen; Curran, James; and Osborne, Miles. 2003. "Bootstrapping POS-taggers using unlabelled data". *Proceedings of the seventh conference on natural language learning at HLT-NAACL 2003*, 49–55. Edmonton: ACL.

Cole, Ronald A.; Mariani, Joseph; Uszkoreit, Hans; Zaenen, Annie; and Zue, Victor (eds). 1996. *Survey of the state of the art in human language technology*. Cambridge: Cambridge University Press.

Creutz, Mathias. 2003. "Unsupervised segmentation of words using prior distributions of morph length and frequency". *Proceedings of the 41st annual meeting of the Association for Computational Linguistics*. Sapporo: ACL.

Creutz, Mathias; and Lagus, Krista. 2002. "Unsupervised discovery of morphemes". *Morphological and phonological learning: Proceedings of the 6th workshop of the ACL special interest group in computational phonology (SIGPHON)*, 21–30. Philadelphia: ACL.

Cucerzan, Silviu; and Yarowsky, David. 2002. "Bootstrapping a multilingual part-of-speech tagger in one day". *Proceedings of CoNLL-2002*. Taipei: ACL.

De Schryver, Gilles-Maurice. 2002. "Web for/as corpus: A perspective for the African languages". *Nordic Journal of African Studies*, 11(2): 266–282.

Ejerhed, Eva; and Källgren, Gunnel. 1997. Stockholm Umeå Corpus (SUC) version 1.0. Research report. Department of Linguistics, Umeå University.

Elworthy, David. 1994. "Does Baum-Welch re-estimation help taggers?" *Fourth conference on applied natural language processing*, 53–58. Stuttgart: ACL.

Engelson, Sean P.; and Dagan, Ido. 1996. "Sample selection in natural language learning". In: Wermter, Stefan; Riloff, Ellen; and Scheler, Gabriele (eds), *Connectionist, statistical and symbolic approaches to learning for natural language processing*, 230–245. Berlin: Springer.

Garvin, Paul. 1967. "The automation of discovery procedure in linguistics". *Language*, 43(1): 172–178.

Ghani, Rayid; Jones, Rosie; and Mladenić, Dunja. 2001a. Building minority language corpora by learning to generate web search queries. Technical Report CMU-CALD-01-100. Carnegie Mellon University Center for Automated Learning and Discovery.

Ghani, Rayid; Jones, Rosie; and Mladenić, Dunja. 2001b. "Mining the web to create minority language corpora". *Proceedings of the Tenth International Conference on Information and Knowledge Management (CIKM 2001)*.

Goldsmith, John. 2001. "Unsupervised learning of the morphology of a natural language". *Computational Linguistics*, 27(2): 153–198.

Harris, Zellig S. 1970. *Papers in structural and transformational linguistics*. Dordrecht: Reidel.

Johnson, Howard; and Martin, Joel. 2003. "Unsupervised learning of morphology for English and Inuktitut". *Companion volume of the proceedings of HLT-NAACL 2003 - short papers*. Edmonton: ACL.

Jurafsky, Daniel; and Martin, James H. 2000. *Speech and language processing. An introduction to natural language processing, computational linguistcis, and speech recognition*. Upper Saddle River: Prentice Hall.

Keller, Bill; and Lütz, Rudi. 1997a. "Learning stochastic context-free grammars from corpora using a genetic algorithm". *Proceedings ICANNGA-97*.

Keller, Bill; and Lütz, Rudi. 1997b. "Evolving stochastic context-free grammars from examples using a minimum description length principle". *Proceedings ICML-97*.

Knuth, Donald E. 1973. *The art of computer programming. Volume 3: Sorting and searching*. Reading, Mass.: Addison-Wesley.

Kroskrity, Paul V.; and Reynolds, Jennifer F. 2001. "On using multimedia in language renewal. Observations from making the CD-ROM *Taitaduhaan*". In: Hinton, Leanne and Hale, Ken (eds), *The green book of language revitalization in practice*, 316–329. San Diego: Academic Press.

Kruskal, Joseph B. 1983. "An overview of sequence comparison". In: Sankoff, David; and Kruskal, Joseph B. (eds), *Time warps,*

string edits, and macromolecules: The theory and practice of sequence comparison, 1–44. Reading, Mass.: Addison-Wesley.

Landauer, T. K.; Foltz, P. W.; and Laham, D. 1998. "Introduction to latent semantic analysis". *Discourse Processes* 25: 259–284.

Lee, Young-Suk; Papineni, Kishore; Roukos, Salim; Emam, Ossama; and Hassan, Hany. 2003. "Language model based Arabic word segmentation". *Proceedings of the 41st annual meeting of the Association for Computational Linguistics*. Sapporo: ACL.

Manandhar, S.; Dzeroski, S.; and Erjavec, T. 1998. "Learning multilingual morphology with CLOG". *Inductive logic programming: Proceedings of the 8th international conference (ILP '98)*, 135–144. Madison, Wis.

Manning, Christopher D. 1998. "The segmentation problem in morphology learning". *NeMLaP3/CoNLL98 workshop on paradigms and grounding in language learning*, 299–305. ACL.

Manning, Christopher D.; and Schütze, Hinrich. 1999. *Foundations of statistical natural language processing*. Cambridge, MA: MIT Press.

Martin, Joel; Johnson, Howard; Farley, Benoit; and Maclachlan, Anna. 2003. "Aligning and using an English-Inuktitut parallel corpus". *Proceedings of the HLT-NAACL 2003 workshop on building and using parallel texts: Data driven machine translation and beyond*. Edmonton: ACL.

Maynard, Diana; Tablan, Valentin; and Cunningham, Hamish. 2003. "NE recognition without training data on a language you don't speak". *Proceedings of the ACL 2003 workshop on multilingual and mixed-language named entity recognition*. Sapporo: ACL.

McEnery, Tony; and Wilson, Andrew. 2001. *Corpus linguistics*. 2nd ed. Edinburgh: Edinburgh University Press.

Merialdo, Bernard. 1994. "Tagging English text with a probabilistic model". *Computational Linguistics*, 20(2): 155–171.

Merkel, Magnus. 1999. *Understanding and enhancing translation by parallel text processing*. (Linköping studies in science and technology, dissertation no. 607). Dept. of Computer and Information Science, Linköping University.

Mitchell, Tom M. 1997. *Machine learning*. New York: McGraw-Hill.

Mitkov, Ruslan (ed.). 2003. *The Oxford handbook of computational linguistics*. Oxford: Oxford University Press.

Mooney, Raymond J. 2003. "Machine learning". In: Mitkov, Ruslan (ed.), *The Oxford handbook of computational linguistics*, 376–394. Oxford: Oxford University Press.

Nelson, Gerard; Wallis, Sean; and Aarts, Bas. 2002. *Exploring natural language: Working with the British component of the International Corpus of English*. Amsterdam: John Benjamins.

Neuvel, Sylvain; and Fulop, Sean A. 2002. "Unsupervised learning of morphology without morphemes". *Morphological and*

phonological learning: Proceedings of the 6th workshop of the ACL special interest group in computational phonology (SIGPHON), 31–40. Philadelphia: ACL.

Ngai, Grace; and Yarowsky, David. 2000. "Rule writing or annotation: Cost-efficient resource usage for base noun phrase chunking". *Proceedings of the 38th annual meeting of the association for computational linguistics*. ACL.

Nilsson, Kristina. 2003. A meta search approach to locating and classifying reading material for learners of Nordic languages. Master's Thesis in Computational Linguistics, Dept. of Linguistics, Uppsala University.

Nilsson, Kristina; and Borin, Lars. 2002. "Living off the land: The web as a source of practice texts for learners of less prevalent languages". *LREC 2002. Third international conference on language resources and evaluation. Proceedings*, 411–418. Las Palmas: ELRA.

Oard, Douglas W.; Doermann, David; Dorr, Bonnie; He, Daqing; Resnik, Philip; Weinberg, Amy; Byrne, William; Khudanpur, Sanjeev; Yarowsky, David; Leuski, Anton; Koehn, Philipp; and Knight, Kevin. 2003. "Desperately seeking Cebuano". *Companion volume of the proceedings of HLT-NAACL 2003 - short papers*. Edmonton: ACL.

Oflazer, Kemal; McShane, Marjorie; and Nirenburg, Sergei. 2001. "Bootstrapping morphological analyzers by combining human elicitation and machine learning". *Computational Linguistics* 27(1): 59–85.

Oflazer, Kemal; and Nirenburg, Sergei. 1999. "Practical bootstrapping of morphological analyzers". *Proceedings of language learning workshop 1999*. ACL.

Ostler, Nicholas. (n.d.). Review: Workshop on language resources for european minority languages, Granada, Spain; 27 May 1998 (morning). <http://www.cstr.ed.ac.uk/~briony/SALTMIL/review.html>. (Accessed on 10 June 2003)

Phillipson, Robert. 2003. *English-only Europe? Challenging language policy*. Andover: Routledge.

Rissanen, Jorma. 1978. "Modeling by shortest data description". *Automatica* 14: 465–471.

Rissanen, Jorma. 1989. *Stochastic complexity in statistical inquiry*. Singapore: World Scientific Publishing Co.

Rogati, Monica; McCarley, Scott; and Yang, Yiming. 2003. "Unsupervised learning of Arabic stemming using a parallel corpus". *Proceedings of the 41st annual meeting of the Association for Computational Linguistics*. Sapporo: ACL.

Sarasola, Kepa. 2000. "Strategic priorities for the development of language technology in minority languages". *LREC 2000.*

Second international conference on language resources and evaluation. Workshop proceedings. Developing language resources for minority languages: Reusability and strategic priorities, 106–109. Athens: ELRA.

Saxena, Anju; and Borin, Lars. (eds) 2006. *Lesser-known languages of South Asia: Status and policies, case studies and applications of language technology*. Berlin: Mouton de Gruyter.

Schafer, Charles; and Yarowsky, David. 2002. "Inducing translation lexicons via diverse similarity measures and bridge languages". *Proceedings of CoNLL-2002*. Taipei: ACL.

Schone, Patrick; and Jurafsky, Daniel. 2000. "Knowledge-free induction of morphology using Latent Semantic Analysis". *Proceedings of CoNLL-2000 and LLL-2000*, 67–72. Lisbon: ACL.

Schone, Patrick; and Jurafsky, Daniel. 2002. "Knowledge-free induction of inflectional morphologies". Proceedings of the second meeting of the North American chapter of the Association for Computational Linguistics.

Sharma, Utpal; Kalita, Jugal; and Das, Rajib. 2002. "Unsupervised learning of morphology for building lexicon for a highly inflectional language". *Morphological and phonological learning: Proceedings of the 6th workshop of the ACL special interest group in computational phonology (SIGPHON)*, 1–10. Philadelphia: ACL.

Snover, Matthew 2002. An unsupervised knowledge free algorithm for the learning of morphology in natural languages. Washington University MS Thesis.

Snover, Matthew G.; and Brent, Michael R. 2001. "A Bayesian model for morpheme and paradigm identification". *Proceedings of the 39th annual meeting of the Association for Computational Linguistics*, 482–490. ACL.

Snover, Matthew G.; and Brent, Michael R. 2002 "A probabilistic model for learning concatenative morphology". *Proceedings of NIPS 2002*. NIPS Foundation.

Snover, Matthew G; Jarosz, Gaja E.; and Brent, Michael R. 2002. "Unsupervised learning of morphology using a novel directed search algorithm: Taking the first step". *Morphological and phonological learning: Proceedings of the 6th workshop of the ACL special interest group in computational phonology (SIGPHON)*, 11–20. Philadelphia: ACL.

Sparck Jones, Karen. 1996. "How much has information technology contributed to linguistics?". Presentation at the *British Academy Symposium on Information Technology and Scholarly Disciplines*, 18–19 October 1996. The page references in the text are to the electronic version available via <http://xxx.lanl.gov/cmp-lg/9702011/>.

Steedman, Mark; Hwa, Rebecca; Clark, Stephen; Osborne, Miles; Sarkar, Anoop; Hockenmaier, Julia; Ruhlen, Paul; Baker, Steven; and Crim, Jeremiah. 2003. "Example selection for bootstrapping statistical parsers". *Proceedings of HLT-NAACL 2003*, 236–243. Edmonton: ACL.

Strassel, Stephanie; Maxwell, Mike; and Cieri, Christopher. 2003. Linguistic resource creation for research and technology development: A recent experiment. *ACM Transactions on Asian Language Processing* 2 (2): 101–117.

Theron, Pieter; and Cloete, Ian. 1997. "Automatic acquisition of two-level morphological rules". *Fifth conference on applied natural language processing*, 103–110. Washington: ACL.

Tiedemann, Jörg. 2003. *Recycling translations – extraction of lexical data from parallel corpora and their application in natural language processing*. (Studia Linguistica Upsaliensia 1). Uppsala: Acta Universitatis Upsaliensis

Trosterud, Trond. 2002. "Parallel corpora as tools for investigating and developing minority languages". In: Borin, Lars (ed.), *Parallel corpora, parallel worlds*, 111–122. Amsterdam: Rodopi.

Vuorela, Katri; and Borin, Lars. 1998. "Finnish Romani". In: Ó Corráin, Ailbhe; and Mac Mathúna, Seamus (eds), *Minority languages in Scandinavia, Britain and Ireland*, 51–76. Uppsala: A&W.

Watson, Bruce W.; and Daciuk, Jan. 2003. "An efficient incremental DFA minimization algorithm". *Natural Language Engineering* 9 (1): 49–64.

Yarowsky, David; and Ngai, Grace. 2001. "Inducing multilingual POS taggers and NP bracketers via robust projection across aligned corpora". *Second meeting of the North American Chapter of the Association for Computational Linguistics*. ACL.

Yarowsky, David; Ngai, Grace; and Wicentowski, Richard. 2001. "Inducing multilingual text analysis tools via robust projection across aligned corpora". *Proceedings of the first international conference on human language technology research*. ACL.

Yarowsky, David; and Wicentowski, Richard. 2000. "Minimally supervised morphological analysis by multimodal alignment". *Proceedings of the 38th annual meeting of the Association for Computational Linguistics*. ACL.

Appendix: Summary of language technology resources and tools

For convenience, in this appendix I present in one place the language technology resources and tools discussed in sections 2.1 – 2.6 above. LDC LoDL survey criteria (see section 2) are in square brackets.

0. Prerequisites
• [language written]
• [standard digital encoding]
• [words separated in writing]
• [simple orthography]
• [sentence punctuation]
• [simple morphology]
• [(existence of) dictionary]
• [(existence of) newspaper]
• [(existence of) Bible (translation)]

1. Foundations
• Corpus: collections of raw text (untagged) [100 000 words of news text]
• Text corpus proper (untagged)
• [100 000 words of parallel text]
• Lexicon: Raw lists of forms, lemmas and affixes
• Machine-readable dictionaries (monolingual, bilingual, thesaurus, other) [10 000 word translation dictionary]
• Morphology: Description and formalization of morphological phenomena
• Speech databases (collections of digitized speech)
• Formal description and dictionaries of units for speech synthesis

2. Basic resources and tools
• Statistical tools for corpus treatment (bigram and trigram frequencies, word counts, collocations ...)
• Part-of-speech (POS) tagger
• POS-tagged corpora
• Lexical database containing information about parts of speech and morphology
• Morphological analyzer/generator [morphological analyzer]
• Speech recognition systems recognizing isolated words

3. Medium-complexity resources and tools
• Environment for (available) tool integration – using a standard for representation of linguistic knowledge: XML/SGML, etc.
• Spelling checker and corrector
• Structured lexical databases including multiword lexical units
• Surface syntax analyzer ("chunker") recognizing simple (nonrecursive) constituents and phrases (NP, PP, verb)
• Web crawler managing language X

4. Advanced resources and tools
• Syntactically annotated corpora ("treebanks")
• Lemmatizer
• Grammar and style checkers
• Integration of dictionaries in text editors
• Lexical-semantic knowledge base. Concept taxonomy, e.g. *WordNet*
• Word sense disambiguation
• Speech processing at sentence level

5. Multilinguality and general applications
• Semantically annotated corpora
• Information retrieval and extraction
• Machine translation systems; translation of NPs and simple sentences
• Dialog systems
• Multilingual lexical-semantic knowledge base
• Language learning systems using human language technology