Linguistic **I**ssues in **L**anguage **T**echnology – **LiLT**
Submitted, October 2011

On Achieving and Evaluating Language-Independence in NLP

Emily M. Bender

Published by CSLI Publications

LiLT volume 6, issue 3 October 2011

On Achieving and Evaluating Language-Independence in NLP

Emily M. Bender, *University of Washington*

1 Introduction

Language independence is commonly presented as one of the advantages of modern, machine-learning approaches to NLP, and it is an important type of scalability. If technology developed for one language can be ported to another merely by amassing appropriate training data in the second language, then the effort put into the development of the technology in the first language can be leveraged to more efficiently create technology for other languages. In cases where the collection of training data represents minimal effort (compared to the algorithm development), this can be very efficient indeed.

In this position paper, I critically review the widespread approaches to achieving and evaluating language independence in the field of computational linguistics and argue that, on the one hand, we are not truly evaluating language independence with any systematicity and on the other hand, that truly language-independent technology requires more linguistic sophistication than is the norm. The rest of the paper is structured as follows: In §2, I motivate the interest of language independence in NLP systems, explore how it is standardly pursued, and make recommendations for how it could be done better, by leveraging the results of linguistic typology. In §3, I survey the papers from ACL2008: HLT and EACL 2009 to give a picture of how language independence is currently evaluated in our field. §4 presents a quick check-list of rec-

LiLT Volume 6, Issue 3, October 2011.
On Achieving and Evaluating Language-Independence in NLP.

ommendations for more typologically-informed NLP, in etiquette-book style. Finally, §5 reviews some work which goes against the trend and explicitly addresses the intersection between computational linguistics and linguistic typology.

2 Language independence: Why and how

2.1 Why language independence?

Truly language-independent NLP technology would be very valuable from both practical and scientific perspectives. From a practical perspective, it would enable more cost-efficient creation of NLP technology across many different language markets as well as more time-efficient creation of applications in situations which require quick ramp-up (see, e.g., the DARPA Surprise Language Exercise from 2003 (Oard, 2003) or the scramble to produce English–Haitian Creole translation systems[1]). In addition, language independence means that technology is more likely to be deployed for languages that have less economic clout. NLP technology for so-called low-density languages also has scientific interest: As argued by Bender and Langendoen (2010), computational methods have much to offer the enterprise of linguistic analysis, but many of these computational methods rely on the availability of other NLP resources, such as POS taggers or parsers. Finally, in the ideal scenario, language-independent NLP systems can teach us something about the nature of human language, and what human languages share in common.

Thus there are practical and scientific reasons to be interested in creating technology that scales from one to many languages. A related question that doesn't get asked too often is, which languages? There are several possible answers to this question, but only some of them make sense. Beginning with those that don't, we have "All logically possible sets of string-meaning pairs" and "All possible sets of string-meaning pairs that could be used as communication systems". The reasonable answers include things like "All currently spoken human languages" (of which Ethnologue lists 6,909[2]), "All languages spoken in X country/continent", "All languages spoken by at least N people", and "All languages with established writing systems". Two points emerge here: The first is that when we claim language independence, we should specify over which set we are claiming independence. The second is that, as soon as we restrict our attention to actually existing human lan-

[1] See e.g. http://research.microsoft.com/en-us/news/features/haitiancreole-020410.aspx (accessed 3/30/10).

[2] http://www.ethnologue.org, accessed 7/23/09

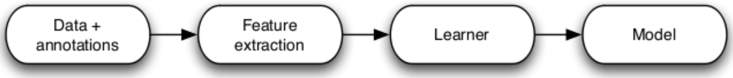

FIGURE 1 Schematic diagram of machine learning

guages (or some subset thereof), we are working with the same domain as linguistic typologists. The results of linguistic typology describe the range of variation in human languages (and tendencies of co-variation across properties of human language) and can be used to inform NLP systems. This idea is taken up in §2.3.

2.2 How is language independence currently pursued?

In papers published at ACL, EACL and similar venues, it is common to assert that the methods presented "apply" to other languages (in some cases with the hedge that certain kinds of resources—e.g., a WordNet—are required). While this is often true, it is not the case that "apply" entails "work". That is, just because the software could be run over training and test data from a different language,[3] doesn't mean that it will work equally or even reasonably well. And if it doesn't work reasonably well, then it is not truly language independent.

At the same time, much is often made of the lack of linguistic knowledge encoded in the algorithms. Figure 1 presents a very schematic diagram of machine learning approaches to language: The diagram begins with annotated data, which is fed into a feature extraction program, resulting in features which are fed to the learner, which in turn produces a model. When the annotations are for example just word boundaries and the features are n-gram counts, or when the annotations are tree or dependency structures using notions which we believe could be applied to any language and the features are arbitrary subparts of those trees, then arguably these algorithms are language-independent in the sense that they have not been created on the basis of any particular explicit analysis of the language at hand. In other words, they could be "applied" without great effort to any other language.

However, the lack of explicitly encoded linguistic knowledge does not ensure that the approach has not been tuned to the development language. Consider for example the case of word-based n-gram models. On the face of it, n-gram models code in no linguistic knowledge. They treat natural language text as simple sequences of symbols and auto-

[3] In some cases, this would require preprocessing, e.g., sentence- or word-boundary detection, which can be non-trivial (cf. e.g. Khudanpur, 2006, §6.3.2).

matically reflect the “hidden” structure through the way it affects the distributions of words in various (flat, unstructured) contexts. However, the effectiveness of n-gram models in English (and similar languages) is partially predicated on two properties of those languages: relatively low levels of inflectional morphology, and relatively fixed word order.

As is well-known by now (Khudanpur, 2006, inter alia), languages with more elaborate morphology (more morphemes per word, more distinctions within the same number of morphological slots, and/or fewer uninflected words) present greater data sparsity problems for n-gram language models. This data sparsity limits the ability of n-gram models to capture the dependencies between open-class morphemes, but also between closed-class morphemes. The information expressed by short function words in English is typically expressed by the inflectional morphology in languages with more elaborate morphological systems. Word-based n-gram models have no way of representing the function morphemes in such a language. In addition, for n-gram models to capture inter-word dependencies, both words have to appear in the n-gram window. This will happen more consistently in languages with relatively fixed word order, as compared to languages with relatively free word order.[4]

Thus even though word-based n-grams models can be built without any hand-coding of linguistic knowledge, they are not truly language independent. Rather, their success depends on typological properties of the languages they were first developed for. A more linguistically-informed (and thus more language independent) approach to n-gram models is the factored language model approach of Bilmes and Kirchhoff (2003). Factored language models address the problems of data-sparsity in morphologically complex languages by representing words as bundles of features, thus capturing dependencies between sub-word parts of adjacent words.

A second example of subtle language dependence comes from Dasgupta and Ng (2007), who present an unsupervised morphological segmentation algorithm meant to be language-independent. Indeed, this work goes much further towards language independence than is the norm (see Section 3). It is tested against data from English, Bengali, Finnish and Turkish, a particularly good selection of languages in that it includes diversity along a key dimension (degree of morphological complexity), as well as representatives of three language families (Indo-

[4] Khudanpur (2006) argues, however, that free word order isn’t as much of a problem at it might appear to be, because local order (within phrases) is relatively stable even when global order (of major sentence constituents) is fluid in the languages studied so far.

European, Uralic, and Altaic). Furthermore, the algorithm is designed to detect more than one prefix or suffix per word, which is important for analyzing morphologically complex languages. However, it seems unrealistic to expect a one-size-fits-all approach to be achieve uniformly high performance across varied languages, and, in fact, it doesn't. Though the system presented in Dasgupta and Ng 2007 outperforms the best systems in the 2006 PASCAL challenge for Turkish and Finnish, it still does significantly worse on these languages than English (F-scores of 66.2 and 66.5, compared to 79.4).

This seems to be due to an interesting interaction of at least two properties of the languages in question. First, the initial algorithm for discovering candidate roots and affixes relies on the presence of bare, uninflected roots in the training vocabulary, extracting a string as a candidate affix (or sequence of affixes) when it appears at the end (or beginning) of another string that also appears independently. In Turkish and Finnish, verbs appear as bare roots in many fewer contexts than in English.[5] This is also true in Bengali, and the authors note that their technique for detecting allomorphs is critical to finding "out-of-vocabulary" roots (those unattested as stand-alone words) in that language. However, the technique for finding allomorphs assumes that "roots exhibit the character changes during attachment, not suffixes" (p.160), and this is where another property of Finnish and Turkish becomes relevant: Both of these languages exhibit vowel harmony, where the vowels in many suffixes vary depending on the vowels of the root, even if consonants intervene. Thus I speculate that at least some of the reduced performance in Turkish and Finnish is due to the system not being able to recognize variants of the same suffixes as the same, and, in addition, not being able to isolate all of the roots.

Of course, in some cases, one language may represent, in some objective sense, a harder problem than another. For example, the difficulty of learning the gender classification of nouns depends on the number of genders to classify the nouns into, as well as the reliability of phonological cues to gender (e.g., word endings) in the language at hand.[6] Another clear example is English letter-to-phoneme conversion, which, as a result of the lack of transparency in English orthography, is a harder problem that letter-to-phoneme conversion in other lan-

[5] In Finnish, depending on the verb class, the bare root may appear in negated present tense sentences, in second-person singular imperatives, and third-person singular present tense, or not at all (Karlsson and Chesterman, 1999). In Turkish, the bare root can function as a familiar imperative, but other forms are inflected (Lewis, 1967, Underhill, 1976).

[6] Thanks to Jeremy Nicholson for pointing out this type of example.

guages. Not surprisingly, the letter-to-phoneme systems described in e.g. Jiampojamarn et al. 2008 and Bartlett et al. 2008 do worse on the English test data than they do on German, Dutch, or French. On the other hand, just because one language may present a harder problem than the other doesn't mean that system developers can assume that any performance differences can be explained in such a way. If one aims to create a language-independent system, then one must explore the possibility that the system includes assumptions about linguistic structure which do not hold up across all languages.

The conclusions I would like to draw from these examples are as follows: A truly language-independent system works equally (or nearly equally) well across languages. When a system that is meant to be language independent does not in fact work equally well across languages, it is likely because something about the system design is making implicit assumptions about language structure. These assumptions are typically the result of "overfitting" to the original development language(s). Here I use the term "overfitting" metaphorically, to call out the way in which, as the developers of NLP methodology, we rely on our intuitions about the structure of the language(s) we're working with and the feedback we get by testing our ideas against particular languages. Feature design is inspired by a kind of tacit linguistics—our knowledge of familiar languages—and success (and failure) on development and test sets from specific familiar languages drives the research process. In the next subsection, I will argue that the best way to achieve language independence is by using explicit, rather than tacit, linguistics and by including, rather than eschewing, linguistic knowledge.

2.3 How should language independence be pursued?

Typically, when we think of linguistic knowledge-based NLP systems, what comes to mind are complicated, intricate sets of language-specific rules. While I would be the last to deny that such systems can be both linguistically interesting and the best approach to certain tasks (cf. Uszkoreit 2002), my purpose here is to point out that there are other kinds of linguistic knowledge that can be fruitfully incorporated into NLP systems. In particular, the results of linguistic typology represent a rich source of knowledge that, by virtue of being already produced by typologists, can be relatively inexpensively incorporated into NLP systems.

Linguistic typology is an approach to the scientific study of language which was pioneered in its modern form by Joseph Greenberg in the

1950s and 1960s (see e.g. Greenberg, 1963).[7] In the intervening decades, it has evolved from a search for language universals and the limits of language variation to what Bickel (2007) characterizes as the study of "what's where why". That is, typologists are interested in how variations on particular linguistic phenomena are distributed throughout the world's languages, both in terms of language families and geography, and how those distributions came to be the way they are.

For the purposes of improving language-independent NLP systems, we are primarily concerned with "what" and "where": Knowing "what" (how languages can vary) allows us to both broaden and parameterize our systems. Knowing "where" also helps with parameterizing, as well as with selecting appropriate samples of languages to test the systems against. We can broaden our systems by studying what typologists have to say about our initial development languages, and identifying those characteristics we might be implicitly relying on. This is effectively what Bilmes and Kirchhoff (2003) did in generalizing n-gram language models to factored language models. We can parameterize our systems by identifying and specifically accommodating relevant language types ("what") and then using databases produced by typologists to map specific input languages to types ("where").[8]

As noted in §2.1, the practical point of language independence is not to be able to handle in principle any possible language in the universe (human or extraterrestrial!), but to improve the scalability of NLP technology across the existing set of human languages. There are approximately 7,000 languages spoken today, of which 347 have more than 1 million speakers.[9] An NLP system that uses different parameters or algorithms for each one of a set of known languages is not language independent. One that uses different parameters or even algorithms for different language *types*, and includes as a first step the classification of the input language, either automatically or with reference to some external typological database, *is* language independent, at least in the relevant, practical sense.

The preeminent typological database among those which are currently publicly available is WALS: The World Atlas of Linguistic Struc-

[7] See Ramat to appear for discussion of much earlier approaches.

[8] In the case of systems working with language pairs, the relevant question could be how typologically similar the two languages are. Cromierès and Kurohashi (2009), for example, suggest that the information added by using parsing as a component of statistical MT systems is more important when the source and target languages have very different syntactic structures.

[9] http://wwww.ethnologue.com/ethno_docs/distribution.asp; accessed 6 February 2009

tures Online (Haspelmath et al., 2008).[10] WALS currently includes 142 chapters studying linguistic features, each of which defines a dimension of classification, describes values along that dimension, and then classifies a large sample of languages. It is also possible to view the data on a language-by-language basis. These chapters represent concise summaries, as well as providing pointers into the relevant literature for more information.

To give a sense of how this information might be of relevance to NLP or speech systems, here is a brief overview of three chapters:

Maddieson (2008) studies tone, or the use of pitch to differentiate words or inflectional categories. He classifies languages into those with no tone systems, those with simple tone systems (a binary contrast between high and low tone), and those with more complex tone systems (more than two tone types). Nearly half of the languages in the sample have some tone, and Maddieson points out that the sample in fact underestimates the number of languages with tone. Information about the presence or absence (and nature) of a tone system has obvious implications for speech processing applications (cf. Kirchhoff 2006, §2.2.2).

Dryer (2008b) investigates prefixing and suffixing in inflectional morphology, looking at 10 common types of affixes (from case affixes on nouns to adverbial subordinator affixes on verbs), and using them to classify languages in terms of tendencies towards prefixing or suffixing.[11] His resulting categories are: little affixation, strongly suffixing, weakly suffixing, equal prefixing and suffixing, weakly prefixing, and strongly prefixing. The most common category (382/894 languages) is predominantly suffixing. Information about the degree of affixation in a language and where in the word affixes tend to appear is useful for lemmatizers, morphological analyzers, bag-of-words approaches to information retrieval, and many other tasks.

Dryer (2008a) investigates the expression of clausal negation. One finding of note is that all languages studied use dedicated morphemes to express negation. This contrasts with the expression of yes-no questions which can be handled with word order changes, intonation, or no overt mark at all. The types of expression of clausal negation that Dryer identifies are: negative affix, negative auxiliary verb, and negative particle. In addition, some languages are classified as using a negative word that may be a verb or may be a particle, as having variation between negative affixes and negative words, and as having double (or

[10] Available online at: http://wals.info

[11] For the purposes of this study, he sets aside less common inflectional strategies such as infixing, tone changes, and stem changes.

two-part) negation, where each negative clause requires two markers, one before the verb, and one after it. Negation, and in particular, the detection of negation in running text is of great interest in biomedical NLP (e.g. Chapman et al. 2001), sentiment analysis, and other meaning-extracting applications. Clausal negation is not the only kind of negation. Nonetheless, knowing how to find it crosslinguistically can be useful.

These examples illustrate several useful aspects of the knowledge systematized by linguistic typology: First, languages show variation beyond that which one might imagine looking only at a few familiar (and possibly closely related) languages. Second, however, that variation is still bounded: Though typologists are always interested in finding new categories that stretch the current classification, for the purposes of computational linguistics, we can get very far by assuming the known types exhaust the possibilities. Finally, because of the work done by field linguists and typologists, this knowledge is available as high-level generalizations about languages, of the sort that can inform the design of linguistically-sophisticated, language-independent NLP systems. Furthermore, even if typological information about a particular language is incomplete, we can often estimate some of the missing values probabilistically on the basis of what is known about the language and typological implications (see §5.1).

Typological information can be incorporated into NLP systems in several different ways: (i) Understanding the typological properties of familiar languages, and how other languages can vary on these same dimensions, can help inform the search for hidden assumptions about language structures in NLP systems. (ii) For some applications, it might make sense to create a range of models and/or include parameters to tune the system to different typological properties. These properties could then be detected or alternatively looked up as a preprocessing step. (iii) Typological knowledge can also be used directly in otherwise unsupervised machine learning systems (such as in Schone and Jurafsky 2001, discussed in §5 below), by taking advantage of known tendencies towards covariation across linguistic dimensions.

2.4 Summary

This section has framed language-independence as a valuable goal in computational linguistics as well as a potential benefit of using machine-learning methods, and argued that the best way to achieve it is to directly consider how the typological properties of the development languages might have informed the design of the system and that the field of linguistic typology is a rich source of information for computational linguists looking to extend the cross-linguistic scalability of their systems. In the next section, I will consider how we can improve our evaluation of the language independence.

3 Evaluating language independence: A report card

As discussed in §2.2, even systems built without any explicit language-specific knowledge can rely for their success on typological properties of the development languages. It follows that we can't deduce language-independence from algorithm design, but rather must prove it with appropriate evaluation methodologies. Ideally, this would involve held-out test languages, with divergent typological properties and drawn from different language families, but this is exceedingly rare, appearing in only 2 of the 217 papers surveyed here: Xia et al. 2009 and Davidov and Rappoport 2009. Short of every paper considering multiple, diverse languages, the field as a whole should be considering a wide range of languages. In order to get a sense of how well we meet this goal, I surveyed the papers from ACL2008: HLT (119 long papers) and EACL 2009 (98 regular papers). Not all of these papers explicitly claim language independence of the methodologies they use (and a few are explicitly language specific), but enough are in the style of knowledge-lean, machine-learning heavy research that it seems fair to evaluate them all together. Indeed, it seems that language independence is often assumed rather than stated, as will be further discussed below.

Table 1 presents the distribution of the papers according to how many different languages or language pairs were studied. The nine papers studying no language pairs were presenting abstract formal proofs regarding grammar formalisms, algorithms for efficient computation of common machine learning paradigms, and the like. More than three-quarters of the papers in each conference looked at exactly one language or language pair (79.8% of the ACL papers and 75.5% of the EACL papers).

The two ACL papers looking at the widest variety of languages were Ganchev et al. 2008 and Nivre and McDonald 2008. Ganchev et al. (2008) explore whether better alignments lead to better translations, across 6 language pairs, in each direction (12 MT systems), collecting data from a variety of sources. Nivre and McDonald (2008) present an approach to dependency parsing which integrates graph-based and transition-based methods, and evaluate the result against the 13 datasets provided in the CoNLL-X shared task (Nivre et al., 2007).

EACL had three papers with very large sets of languages or language

Languages or language	Number of papers	
pairs considered	ACL2008: HLT	EACL 2009
0	3	6
1	95	74
2	13	7
3	3	3
4	2	2
5	1	1
9	0	1
10	0	1
12	1	0
13	1	0
100+	0	3
Total	119	98
Mean	1.41	11.69

TABLE 1 Number of languages/language pairs considered

pairs considered. Davidov and Rappoport (2009) define a task of cross-lingual concept lexicalization, in which a set of words instantiating a concept (e.g., *apple, banana, . . .* for 'fruit') are input in one language, and a set of words describing the same concept are output in another language. This methodology was evaluated against 100 language pairs, involving 46 languages. 45 of the language pairs had English as the source, 45 had English as the target, and 10 had English as neither source nor target. Mukherjee et al. (2009) develop a network-based method to explore the co-occurrence of consonant phonemes across the phoneme inventories of natural languages. They apply this method to the UPSID (Maddieson, 1984) data set (phoneme inventories from 317 languages) and find evidence for the linguistic notions of markedness and implicational universals in phoneme inventories. In this case, the study is not evaluating a claim or technique against multiple languages, but rather exploring typological facts across a sample of 317 languages. Finally, Xia et al. (2009) approach the problem of language identification in the contexts of short linguistic examples harvested from linguistics papers on the web. Their dataset is drawn from the ODIN database (Lewis, 2006), and includes approximately 600 languages.

As so much work in computational linguistics is driven by the availability of data sets, it is encouraging to see multilingual data sets such as those from ODIN and the CoNLL-X shared task become available. The field as a whole will be in a better position to test (and improve)

Language	Studies		Genus	Studies		Family	Studies	
	N	%		N	%		N	%
English	81	63.28	Germanic	91	71.09	Indo-European	109	85.16
German	5	3.91						
Dutch	3	2.34						
Danish	1	0.78						
Swedish	1	0.78						
Czech	3	2.34	Slavic	8	6.25			
Russian	2	1.56						
Bulgarian	1	0.78						
Slovene	1	0.78						
Ukranian	1	0.78						
Portuguese	3	2.34	Romance	8	6.25			
Spanish	3	2.34						
French	2	1.56						
Hindi	2	1.56	Indic	2	1.56			
Arabic	4	3.13	Semitic	9	7.03	Afro-Asiatic	9	7.03
Hebrew	4	3.13						
Aramaic	1	0.78						
Chinese	5	3.91	Chinese	5	3.91	Sino-Tibetan	5	3.91
Japanese	3	2.34	Japanese	3	3.24	Japanese	3	3.24
Turkish	1	0.78	Turkic	1	0.78	Altaic	1	0.78
Wambaya	1	0.78	W. Barkly	1	0.78	Australian	1	0.78
Total	128	100.00		128	100.00		128	100.00

TABLE 2 Languages studied in ACL 2008 papers, by language genus and family

the cross-linguistic applicability of various methods to the extent that more such datasets are produced. It is worth noting, however, that the sheer number of languages tested is not the only important factor: Because related languages tend to share typological properties, it is also important to sample across the known language *families*. The three outlier studies from EACL 2009 continue to fare well even when language families are taken under consideration. That same is not true, however, of the remaining 204 papers: The modest coverage of linguistic diversity is reduced when viewed from this angle.

Tables 2 and 3 tabulate the studies by language, genus and family for ACL2008: HLT and EACL 2009, respectively, for those papers that presented methodologies concerned with producing results in one language at a time.[12] The first thing to note in these tables is the concentration

[12] The very interesting study by Snyder and Barzilay (2008) on multilingual approaches to morphological segmentation was difficult to classify. Their methodology involved jointly analyzing two languages at a time in order to produce morphological segmenters for each. Since the resulting systems were monolingual, the data from these studies are included in Table 2. Conversely, Garera and Yarowsky (2009) and Navigli (2009) use information from German and Italian, respectively, to aid in

Language	Studies		Genus	Studies		Family	Studies	
	N	%		N	%		N	%
English	48	54.54	Germanic	63	71.59	Indo-European	80	90.90
German	6	6.81						
Swedish	4	4.54						
Dutch	3	3.41						
Danish	1	1.14						
Icelandic	1	1.14						
French	3	3.41	Romance	8	9.09			
Spanish	2	2.27						
Italian	1	1.14						
Latin	1	1.14						
Portuguese	1	1.14						
Bengali	2	2.27	Indic	5	5.68			
Hindi	2	2.27						
Urdu	1	1.14						
Czech	2	2.27	Slavic	4	4.54			
Bulgarian	1	1.14						
Slovene	1	1.14						
Arabic	1	1.14	Semitic	3	3.41	Afro-Asiatic	9	3.41
Hebrew	1	1.14						
Tigrinya	1	1.14						
Chinese	2	2.27	Chinese	2	2.27	Sino-Tibetan	5	2.27
Turkish	1	1.14	Turkic	1	1.14	Altaic	1	1.14
Total	88	100.00		88	100.00		88	100.00

TABLE 3 Languages studied in EACL 2009 papers by language genus and family, exclusive of Mukherjee et al. 2009 and Xia et al. 2009

of work on English: 63% of the single-language studies in ACL and 55% of the single-language studies in EACL concerned English.[13] In addition, languages closely related to English are overrepresented, with the Germanic genus accounting for 71-72% of the studies in the two conferences and the Indo-European family as a whole 85% at ACL and 91% at EACL.

Ethnologue[14] lists 94 language families. ACL2008: HLT papers studied seven (the six shown in Table 2, plus Uralic, represented by Finnish in Table 5). EACL papers studied eight (see Tables 3 and 7). Of course, the distribution of languages (and perhaps more to the point, speakers) is not uniform across language families. Table 4 gives the five most populous language families, again from Ethnologue.[15] These language families together account for almost 85% of the world's population.

Next we turn to the language-pair studies: papers on machine translation, bilingual lexicon construction, transliteration, etc. Tables 5 and 6 catalog the language-pair studies from ACL2008: HLT and EACL 2009 respectively.[16] The EACL table does not include information about the language pairs studied by Davidov and Rappoport (2009). I have chosen to exclude this paper because it is an outlier: the 100 language pairs studied there would swamp the data from the 81 language pairs studied in the other 21 papers tabulated here.

In the EACL data at least, we see more diversity of language family in the language-pair studies than in the single language studies: only 53% of the language pair studies paired English (or French) with an Indo-European language, and a total of seven language families are explored as against four in the single language studies. The tally of language-pair studies by genus and family is shown in Table 7. Nonetheless, one thing jumps out from these tables: the predominance of English. Every language pair in the ACL papers and in every language pair but three in the EACL papers (exclusive of Davidov and Rappoport 2009) involved English on one side or the other. The three EACL studies

a task evaluated in English. Since they don't also evaluate in German and Italian, these studies are included in Table 6 in order to represent the non-English languages.

[13] Mukherjee et al. 2009 and Xia et al. 2009 did not include lists of the languages used, and are not included in these numbers.

[14] http://www.ethnologue.com/ethno_docs/distribution.asp, accessed on 6 February 2009.

[15] Ibid. Example languages are included to give the reader a sense of where these language families are spoken, and are deliberately chosen to represent the breadth of each language family while still being relatively recognizable to a computational linguistics audience.

[16] Tables 6 and 7 include data points from Schroeder et al. 2009, a paper on multisource machine translation.

Language family	Living languages	Examples	% population
Indo-European	430	Welsh Pashto Bengali	44.78
Sino-Tibetan	399	Mandarin Sherpa Burmese	22.28
Niger-Congo	1,495	Swahili Wolof Bissa	6.26
Afro-Asiatic	353	Arabic Coptic Somali	5.93
Austronesian	1,246	Bali Tagalog Malay	5.45
Total	3,923		84.7

TABLE 4 Five most populous language families, from Ethnologue

Source	Target	N	Source	Target	N	Symmetrical pair	N
Chinese	English	9	English	Chinese	2	English, Chinese	3
Arabic	English	5	English	Arabic	2	English, Arabic	1
French	English	2	English	French	2	English, French	1
Czech	English	1	English	Czech	2	English, Spanish	1
Finnish	English	1	English	Finnish	1		
German	English	1	English	German	1		
Italian	English	1	English	Italian	1		
Spanish	English	1	English	Spanish	1		
			English	Greek	1		
			English	Russian	1		

TABLE 5 Language pairs studied in ACL 2008 papers

Source	Target	N	Source	Target	N	Symmetrical pair	N
German	English	3	English	Russian	1	English, French	3
Arabic	English	2	English	Arabic	1	English, Chinese	1
Chinese	English	2	English	Chinese	1	English, Greek	1
French	English	2	English	Dutch	1	English, Japanese	1
Hebrew	English	1	English	Finnish	1	English, Malay	1
Italian	English	1	English	French	1	*French, Italian*	1
Finnish	English	1	English	Hindi	1	*French, Dutch*	1
Russian	English	1	English	Italian	1		
Spanish	English	1	English	Kannada	1		
Swedish	English	1	English	Spanish	1		
French + Swedish	English	1	English	Swedish	1		
French + Spanish	English	1	English	Tamil	1		
French + Portuguese + Danish + Italian	English	1	*Italian*	*French*	1		

TABLE 6 Language pairs studied in EACL 2009 papers, exclusive of Davidov and Rappoport 2009

not involving English instead chose French, which is not typologically very distant from English. By restricting our attention in this way, we are failing to test any methodology against language pairs where both languages are morphologically complex, or both languages have relatively free word order, or both languages show frequent zero anaphora, etc.

In general, this picture is not very encouraging regarding the linguistic diversity of studies in computational linguistics. Perhaps most astonishing, however, is the fact that, of the 45 papers studying only English at EACL, 33 neglected to state directly that English was the language under study.[17] In some cases, it was possible to tell that English was in fact the language in question because the authors cited linguistic resources for English (e.g., (English) WordNet (Fellbaum, 1998) or the (English) Penn Treebank (Marcus et al., 1993)). In others, the only clue was linguistic examples given in English, or statements about how the data was collected (from undergraduates) in combination with the authors' affiliations (US institutions). While the generalizations aren't clear cut, it seems that this tendency is more pronounced in papers on tasks to do with extracting meaning from text (fact extraction, sentiment analysis, and summarization) than it is in papers relating to analysis of language structure directly (parsing, POS tagging, etc.). None of the papers working on bilingual (e.g., MT, bilingual lexicon extraction, etc.) tasks failed to identify the languages being studied. Only one paper that didn't directly identify its languages was work-

[17] This phenomenon was also observed in the ACL papers, but not systematically coded in the survey, so I can only report numbers for EACL here.

	Studies			Studies	
L1, L2 genus	N	%	L1, L2 family	N	%
English, Romance	20	24.69	English, Indo-European	40	49.38
English, Germanic	8	9.88			
English, Slavic	6	7.41			
English, Romance + Germanic	3	3.70			
English, Greek	2	2.47			
English, Indic	1	1.23			
English, Chinese	18	22.22	English, Sino-Tibetan	18	22.22
English, Semitic	12	14.18	English, Afro-Asiatic	12	14.81
English, Finnic	4	4.94	English, Uralic	4	4.94
English, Southern Dravidian	2	2.47	English, Dravidian	2	2.47
English, Japanese	1	1.23	English, Japanese	1	1.23
English, Sundic	1	1.23	English, Austronesian	1	1.23
French, Romance	2	2.47	French, Indo-European	3	3.70
French, Germanic	1	1.23			
Total	81	100.00		81	100.00

TABLE 7 Language pairs studied in ACL 2008 and EACL 2009 papers by genus and family, exclusive of Davidov and Rappoport 2009

ing on languages other than English: Dinarelli et al. (2009) evaluate reranking for spoken language understanding against two corpora, one described as a corpus of Italian, and the other described as being produced by a French project (and thus, presumably, containing French speech).

This lack of specification of the language of study seems to follow from a sense that English is a suitably representative language, combined with the idea that any methodology not explicitly coded to include language-specific knowledge must therefore be language-independent. This idea is presupposed in the writing, so that, much as the avoidance of first-person pronouns in certain styles of scientific text lends an aura of objectivity to the procedures and results, the avoidance of specifying the language being studied gives an aura of language-independence. Both are in fact equally false: the process of doing science centrally involves the subjective perceptions and thought processes of the scientists, and language-independence can only be shown by testing against multiple languages.

To summarize the results of this survey, the field of computational linguistics, with a few important exceptions and as viewed through the papers published in ACL2008: HLT and EACL 2009 is not going very far towards evaluating language-independence of the methodologies we propose and study. It is my hope that the coming years will see the creation of more multilingual data sets, as well as a change in practice towards testing purportedly or potentially language-independent methods against an interesting range of languages. This does not mean that every paper needs to address hundreds of languages or language pairs on the model of Davidov and Rappoport 2009 or Xia et al. 2009. Rather, a small sample of languages would do, if they were interestingly diverse in their typological features, history and geography, and if it wasn't always the same small set of languages over and over again.[18]

4 Prescriptions for Typologically-Informed NLP

This section provides a brief list, in etiquette-book format, of suggestions for producing more typologically-informed (and thus hopefully more language-independent) NLP research. These suggestions are aimed at both authors and reviewers, or rather, at members of the NLP community in both their author and reviewer guises.

Do state the name of the language that is being studied, even if it's English. Acknowledging that we are working on a particular language foregrounds the possibility that the techniques may in fact be language-specific. Conversely, neglecting to state that the particular data used were in, say, English, gives false veneer of language-independence to the work.

Do state the set of languages that the system is meant to generalize to. Is it designed to handle any language with a written tradition? Any human language currently or recently spoken? Indo-European languages? Morphologically simple/complex languages?

Do explicitly note which aspects of the methodology are intended to be language-independent, and which are explicitly language-dependent. This will help reviewers (and eventually readers) understand and evaluate the crosslinguistic applicability of the methods. Those with knowledge of particular languages or linguistics in general will be better able to provide feedback in terms of where the language independence is likely to break down.

Don't require new methods to be tested against the same old languages or language pairs for comparability of results. This kind of tendency in conference reviewing in particular can narrow (and in fact probably already has narrowed) the focus of the field and can stifle work that pushes the boundaries of language independence.

Do evaluate claims of language independence by testing the algorithm against multiple languages. In a truly strict evaluation, we would separate development languages from test languages the way we sepa-

[18] An anonymous reviewer points out that the data that I have collected isn't really longitudinal. It could well be that the diversity found in 2008 and 2009 is an improvement over years past. If so, this is a laudable trend. Regardless, we have a ways to go.

rate development data from test data.

Don't evaluate language independence by only testing against related and/or typologically similar languages. Success of a system developed for one language on data from another shows that the system generalizes somewhat, but to the extent that the languages share typological properties, the evidence of ability to generalize is weak.

Do develop parallel data sets for typologically interesting samples of languages. The more multilingual data sets (with standardized annotations across languages) that are available, the easier it will be for NLP researchers to evaluate claims of language independence. The more these multilingual data sets represent typologically balanced samples, the better these evaluations will be. The data set produced by Nivre et al. (2007) for the 2007 CoNLL shared tasks is a good step in this direction, though the languages included lack genealogical and typological diversity.

Do expect comparable performance across languages from language-independent systems. When performance varies, do error analysis based on typological properties, exploring what it is about the languages that makes the algorithm more successful in one or the other, and how that might relate to assumptions in the algorithm itself.

Do talk to linguists. Linguists can be very helpful in detecting implicit assumptions about linguistic structure in descriptions of algorithms, and their input can be very valuable in the design stage as well as in error analysis (see previous point).

Do make use of typological information. The hard work of collecting this information has already been done (or is being done) by linguists. Making use of it does not make a machine-learning NLP system any less general: typologists are already studying the largest set of languages it makes sense to generalize over.

There are many factors that can make it difficult to follow these recommendations.[19] These include: (i) pressures from funding agencies to work on particular languages of interest, (ii) scarcity of annotated (and curated) data from languages outside the well-studied few and (iii) the importance of shared data sets and shared tasks for comparability across studies. I believe the solution is to be alert for opportunities to extend the range of data sets to languages that are otherwise understudied, and to pursue them when they arise. In addition, as reviewers, we should recognize the value added of working with other languages when we review both papers and grant proposals.

[19] I am grateful to Fei Xia for helpful discussion of these points.

5 Computational Linguistics and Linguistic Typology

This section reviews the small but growing body of literature working at the intersection of computational linguistics and linguistic typology. Some of this work is leveraging results from linguistic typology for applications in computational linguistics. Other studies are applying computational methods to discover typological generalizations. This is potentially useful for typology (giving back), but also for computational linguistics: If it leads to better typological information, that in turn could engender more language-independent NLP systems.

5.1 Computational approaches to typology

The advent of large, publicly available databases of typological properties, in particular WALS (Haspelmath et al., 2008), has opened the door to the application of computational methods to traditional questions of linguistic feature co-occurrence in typology.[20] Two studies in this area are Daumé III and Campbell 2007 and Bakker 2008. Using different statistical modeling techniques, Daumé III & Campbell and Bakker each mine the data in WALS to look for typological implications, i.e., tendencies for languages with one property or set of properties to also have another property. In both cases, the authors discover known typological implications (going back to Greenberg 1963, e.g., verb-object order predicts relative clauses following nouns), as well as new candidates.

Computational techniques can also be applied to the related set of questions of linguistic co-development, i.e., genealogical relationships between languages as well as areal effects, or the influence of languages in contact on one another. Daumé III (2009) uses the feature and geographical data from WALS as input to a Bayesian model designed to discover areal groupings. Taking a somewhat different approach, Nakhleh et al. (2005) build a model (called 'perfect phylogenetic networks') that can accommodate both mutual descent and borrowing across languages, and apply it to data from Indo-European languages to estimate the degree of isolation of the various sub-families in the early development of these languages.

5.2 Leveraging typology in computational linguistics

Looking the other direction, the results of linguistic typology can be integrated into computational linguistics in at least two ways: to inform the design of models meant to be cross-linguistically applicable, or directly as information incorporated into such models. Schutlz and Kirchhoff 2006, a thorough investigation of issues in multilingual speech

[20] Though see Dryer 2009.

processing, nicely illustrates the first approach. The second chapter of that work (Kirchhoff, 2006) provides an overview of the relevant aspects of linguistic typology (and supporting concepts from linguistics) that can impact the way that speech systems work across languages. Chapter 4 (Schultz, 2006) provides an overview of the state-of-the-art in multilingual acoustic modeling, relying on cross-linguistic analysis of phoneme inventories (sound systems) to motivate the design of language-independent and language-adaptive acoustic models. Other chapters address multilingual dictionaries (Adda-Decker and Lamel, 2006), multilingual language modeling (Khudanpur, 2006), and various applications.

Integrating results of linguistic typology as a knowledge source can be useful for both stochastic and knowledge-engineering approaches to NLP. On the stochastic side, Schone and Jurafsky (2001) use typological implications (of the same sort that Daumé III and Campbell (2007) and Bakker (2008) aim to discover) as prior information in a Bayesian approach to inducing class labels in a POS tagging system.[21] One can imagine similarly using typological generalizations to create the prototypes used by Haghighi and Klein (2006) for unsupervised grammar induction. On the knowledge engineering side, the Grammar Matrix project (Bender et al., 2002, Bender and Flickinger, 2005, Drellishak, 2009) has been creating libraries of implemented HPSG analyses which can be added on to a cross-linguistic core grammar through a web-based customization system.[22] These libraries are intended to cover all known variants of each phenomenon, and are therefore based on a thorough review of the relevant typological literature.

6 Conclusion

Building systems that can work for any natural language is a laudable goal for NLP systems, and a plausible benefit of the machine learning approach. I have argued in this paper, that, contrary to popular belief, it is a goal that is not achievable without linguistic knowledge. Fortunately, linguistic typology can help. The set of currently spoken human languages is interestingly large, but not infinite. We know from linguistic typology that the variation is bounded, though greater than you might guess from just one or two languages. In the previous section, I have briefly reviewed how results from linguistic typology are beginning to be incorporated into computational linguistic research.

[21] Unfortunately, this system was only ever tested on English.

[22] The customization system can be accessed here: http://www.delph-in.net/matrix/customize/matrix.cgi

Furthermore, I have argued that if we're serious about language independence as a goal, it needs to be reflected in how we evaluate NLP systems. In particular, we need to broaden the range of languages we work with on a regular basis, sampling carefully to cover an interesting range of typological characteristics and language families. Finally, to truly evaluate language independence, we should include held-out languages as well as held-out data in our evaluations.

An anonymous reviewer suggests that most computational linguists are computer scientists and are not particularly interested in nor attuned to issues of language independence. On the contrary, I would argue that anyone engaged in the scientific analysis of language or linguistic data, with or without the assistance of computers, is by definition a linguist, and therefore responsible for understanding the issues that arise in analyzing natural languages. While it is of course not reasonable to expect everyone working in NLP to have a dual degree, I believe it is reasonable to expect some coursework or other studies in linguistics as well as a high degree of collaboration with researchers fully trained in linguistics.

In this context, it is fair to distinguish between people specializing in NLP and people specializing in machine learning who use NLP as an application area. The former should have some familiarity with linguistics. The latter should at least have some familiarity with linguists. More generally, it is both reasonable and productive to have people specializing in machine learning apply their algorithms to application areas including NLP. In such collaborations, however, the specification of tasks and evaluation metrics should involve subject matter experts, in this case (computational) linguists.

Of course, for these collaborations to work, linguists have to keep up their end of the bargain as well, and not all work in linguistics is sufficiently grounded in data to fit the needs of computational linguists. It is easy to see how a computer scientist could decide that linguistics is not helpful on the basis of a course in mainstream theoretical linguistics. One of the goals of this paper has been to encourage the NLP community to look further into the available subfields of linguistics, and in particular to highlight the potential of the subfield of linguistic typology to provide both very useful information and a perhaps more compatible perspective on linguistic data.

Acknowledgments

This paper is a revised and expanded version of Bender 2009. I am grateful to Stephan Oepen, Timothy Baldwin, Fei Xia, Jeremy Nicholson and two anonymous reviewers for helpful discussion. Any remaining infelicities are my own. This material is based in part upon work supported by the National Science Foundation under Grant No. 0644097. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author and do not necessarily reflect the views of the National Science Foundation.

References

Adda-Decker, Martine and Lori Lamel. 2006. Multilingual dictionaries. In T. Schutlz and K. Kirchhoff, eds., *Multilingual Speech Processing*, pages 123–168. Burlington, MA: Elsevier.

Bakker, Dik. 2008. Linfer: Inferring implications from the wals database. *Sprachtypologie und Universalienforschung* 61.

Bartlett, Susan, Grzegorz Kondrak, and Colin Cherry. 2008. Automatic syllabification with structured SVMs for letter-to-phoneme conversion. In *Proceedings of ACL-08: HLT*, pages 568–576. Columbus, Ohio: Association for Computational Linguistics.

Bender, Emily M. 2009. Linguistically naïve != language independent: Why NLP needs linguistic typology. In *Proceedings of the EACL 2009 Workshop on the Interaction between Linguistics and Computational Linguistics: Virtuous, Vicious or Vacuous?*, pages 26–32. Athens, Greece: Association for Computational Linguistics.

Bender, Emily M. and Dan Flickinger. 2005. Rapid prototyping of scalable grammars: Towards modularity in extensions to a language-independent core. In *Proceedings of the 2nd International Joint Conference on Natural Language Processing IJCNLP-05 (Posters/Demos)*. Jeju Island, Korea.

Bender, Emily M., Dan Flickinger, and Stephan Oepen. 2002. The grammar matrix: An open-source starter-kit for the rapid development of cross-linguistically consistent broad-coverage precision grammars. In J. Carroll, N. Oostdijk, and R. Sutcliffe, eds., *Proceedings of the Workshop on Grammar Engineering and Evaluation at the 19th International Conference on Computational Linguistics*, pages 8–14. Taipei, Taiwan.

Bender, Emily M. and D. Terence Langendoen. 2010. Computational linguistics in support of linguistic theory. *Linguistic Issues in Language Technology* 3:1–31.

Bickel, Balthasar. 2007. Typology in the 21st century: Major current developments. *Linguistic Typology* pages 239–251.

Bilmes, Jeff A. and Katrin Kirchhoff. 2003. Factored language models and generalized parallel backoff. In *in Proceedings of HLT/NACCL, 2003*, pages 4–6.

Chapman, W.W., W. Bridewell, P. Hanbury, G.F. Cooper, and B.G. Buchanan. 2001. A simple algorithm for identifying negated findings and diseases in discharge summaries. *Journal of biomedical informatics* 34(5):301–310.

Cromierès, Fabien and Sadao Kurohashi. 2009. An alignment algorithm using belief propagation and a structure-based distortion model. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 166–174. Athens, Greece: Association for Computational Linguistics.

Dasgupta, Sajib and Vincent Ng. 2007. High-performance, language-independent morphological segmentation. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 155–163. Rochester, New York: Association for Computational Linguistics.

Daumé III, Hal. 2009. Non-parametric Bayesian model areal linguistics. In *North American Chapter of the Association for Computational Linguistics (NAACL)*. Boulder, CO.

Daumé III, Hal and Lyle Campbell. 2007. A Bayesian model for discovering typological implications. In *Conference of the Association for Computational Linguistics (ACL)*. Prague, Czech Republic.

Davidov, Dmitry and Ari Rappoport. 2009. Translation and extension of concepts across languages. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 175–183. Athens, Greece: Association for Computational Linguistics.

Dinarelli, Marco, Alessandro Moschitti, and Giuseppe Riccardi. 2009. Reranking models for spoken language understanding. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 202–210. Athens, Greece: Association for Computational Linguistics.

Drellishak, Scott. 2009. *Widespread But Not Universal: Improving the Typological Coverage of the Grammar Matrix*. Ph.D. thesis, University of Washington.

Dryer, Matthew S. 2008a. Negative morphemes. In M. Haspelmath, M. S. Dryer, D. Gil, and B. Comrie, eds., *The World Atlas of Language Structures Online*. Munich: Max Planck Digital Library. Available online at http://wals.info/feature/112. Accessed on 2009-02-07.

Dryer, Matthew S. 2008b. Prefixing vs. suffixing in inflectional morphology. In M. Haspelmath, M. S. Dryer, D. Gil, and B. Comrie, eds., *The World Atlas of Language Structures Online*. Munich: Max Planck Digital Library. Available online at http://wals.info/feature/26. Accessed on 2009-02-07.

Dryer, Matthew S. 2009. Problems testing typological correlations with the online wals. *Linguistic Typology* 13(1):121 – 135.

Fellbaum, Christiane, ed. 1998. *WordNet: An Electronic Lexical Resource*. Cambridge MA: MIT Press.

Ganchev, Kuzman, João V. Graça, and Ben Taskar. 2008. Better alignments = better translations? In *Proceedings of ACL-08: HLT*, pages 986–993. Columbus, Ohio: Association for Computational Linguistics.

Garera, Nikesh and David Yarowsky. 2009. Structural, transitive and latent models for biographic fact extraction. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 300–308. Athens, Greece: Association for Computational Linguistics.

Greenberg, Joseph. 1963. Some universals of grammar with particular reference to the order of meaningful elements. In *Univerals of Language*, pages 73–113. Cambridge: MIT Press.

Haghighi, Aria and Dan Klein. 2006. Prototype-driven grammar induction. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 881–888. Sydney, Australia: Association for Computational Linguistics.

Haspelmath, Martin, Matthew S. Dryer, David Gil, and Bernard Comrie, eds. 2008. *The World Atlas of Language Structures Online*. Munich: Max Planck Digital Library. http://wals.info.

Jiampojamarn, Sittichai, Colin Cherry, and Grzegorz Kondrak. 2008. Joint processing and discriminative training for letter-to-phoneme conversion. In *Proceedings of ACL-08: HLT*, pages 905–913. Columbus, Ohio: Association for Computational Linguistics.

Karlsson, Fred and Andrew Chesterman. 1999. *Finnish: An Essential Grammar*. London: Routledge.

Khudanpur, Sanjeev P. 2006. Multilingual language modeling. In T. Schutlz and K. Kirchhoff, eds., *Multilingual Speech Processing*, pages 169–205. Burlington, MA: Elsevier.

Kirchhoff, Katrin. 2006. Language characteristics. In T. Schutlz and K. Kirchhoff, eds., *Multilingual Speech Processing*, pages 5–31. Burlington, MA: Elsevier.

Lewis, Geoffrey. 1967. *Turkish Grammar*. Oxford: Clarendon Press.

Lewis, William D. 2006. ODIN: A model for adapting and enriching legacy infrastructure. In *Proceedings of the e-Humanities Workshop, Held in cooperation with e-Science*. Amsterdam.

Maddieson, Ian. 1984. *Patterns of Sounds*. Cambridge: Cambridge University Press.

Maddieson, Ian. 2008. Tone. In M. Haspelmath, M. S. Dryer, D. Gil, and B. Comrie, eds., *The World Atlas of Language Structures Online*. Munich: Max Planck Digital Library. Available online at http://wals.info/feature/13. Accessed on 2009-02-07.

Marcus, Mitchell P., Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics* 19:313–330.

Mukherjee, Animesh, Monojit Choudhury, and Ravi Kannan. 2009. Discovering global patterns in linguistic networks through spectral analysis: A case study of the consonant inventories. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 585–593. Athens, Greece: Association for Computational Linguistics.

Nakhleh, Luay, Don Ringe, and Tandy Warnow. 2005. Perfect phylogenetic networks: A new methodology for reconstructing the evolutionary history of natural languages. *Language* 81(2):382–420.

Navigli, Roberto. 2009. Using cycles and quasi-cycles to disambiguate dictionary glosses. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 594–602. Athens, Greece: Association for Computational Linguistics.

Nivre, Joakim, Johan Hall, Sandra Kübler, Ryan McDonald, Jens Nilsson, Sebastian Riedel, and Deniz Yuret. 2007. The CoNLL 2007 shared task on dependency parsing. In *Proceedings of the CoNLL Shared Task Session of EMNLP-CoNLL 2007*, pages 915–932. Prague, Czech Republic: Association for Computational Linguistics.

Nivre, Joakim and Ryan McDonald. 2008. Integrating graph-based and transition-based dependency parsers. In *Proceedings of ACL-08: HLT*, pages 950–958. Columbus, Ohio: Association for Computational Linguistics.

Oard, Douglas W. 2003. The surprise language exercises. *ACM Transactions on Asian Language Information Processing (TALIP)* 2(2):79–84.

Ramat, Paolo. to appear. The (early) history of linguistic typology. In J. J. Song, ed., *The Oxford Handbook of Linguistic Typology*. Oxford: Oxford University Press.

Schone, Patrick and Daniel Jurafsky. 2001. Language-independent induction of part of speech class labels using only language universals. In *IJCAI-2001 Workshop on Machine Learning: Beyond Supervision*.

Schroeder, Josh, Trevor Cohn, and Philipp Koehn. 2009. Word lattices for multi-source translation. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 719–727. Athens, Greece: Association for Computational Linguistics.

Schultz, Tanja. 2006. Multilingual acoustic modeling. In T. Schutlz and K. Kirchhoff, eds., *Multilingual Speech Processing*, pages 71–122. Burlington, MA: Elsevier.

Schutlz, Tanja and Katrin Kirchhoff, eds. 2006. *Multilingual Speech Processing*. Burlington, MA: Elsevier.

Snyder, Benjamin and Regina Barzilay. 2008. Unsupervised multilingual learning for morphological segmentation. In *Proceedings of ACL-08: HLT*, pages 737–745. Columbus, Ohio: Association for Computational Linguistics.

Underhill, Robert. 1976. *Turkish Grammar*. Cambridge, MA: MIT Press.

Uszkoreit, Hans. 2002. New chances for deep linguistic processing. In *Proceedings of COLING 2002*.

Xia, Fei, William Lewis, and Hoifung Poon. 2009. Language ID in the context of harvesting language data off the web. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 870–878. Athens, Greece: Association for Computational Linguistics.