Book Review

Language and Computers

Markus Dickinson*, Chris Brew‡, and Detmar Meurers¶
(*Indiana University, ‡Educational Testing Service, and ¶University of Tübingen)

Reviewed by
Mats Wirén
Stockholm University

Any textbook on computational linguistics today must position itself relative to Jurafsky and Martin (2009), which, by virtue of its depth and comprehensiveness, is the standard introduction and reference for speech and language processing. Readers without substantial background may find that book too demanding, however, and then *Language and Computers*, by Dickinson, Brew, and Meurers, may be a better choice. Roughly, this is how *Language and Computers* distinguishes itself from Jurafsky and Martin (2009):

1. It does not presuppose any computing background. More specifically, as stated in the Overview for Instructors, the book assumes "no mathematical or linguistic background beyond normal high-school experience." The book certainly has the ambition to explain some of the inner workings of the technology, but does so starting from a general level.

2. It is structured according to major applications (treated in six chapters), with the theoretical material being introduced along the way as it is needed. Jurafsky and Martin (2009) is instead structured according to linguistic description levels, starting with words and gradually proceeding through higher levels (albeit with a final part on applications). The idea in *Language and Computers* is that readers without a background in computing will at least be familiar with real-world tasks in which computers deal with language, and will find a textbook structured in this way more accessible. It also gives the authors an opportunity to show how the underlying techniques recur across different applications.

3. Its topic is processing of text. There is some discussion in the first chapter of the nature and representation of speech, but the applications (with the possible exception of dialogue) are text-oriented.

4. It is introductory, typically aimed at a quarter-length course according to the Overview for Instructors. At 232 pages, it is about one-quarter of the length of Jurafsky and Martin (2009), which has enough material for a full-year course.

Chapter 1 is a prologue with a refreshingly broad perspective, dealing with how written and spoken language are represented in computers. First, the basic writing systems are described: alphabetical (roughly, one character–one phoneme), syllabic (roughly, one character–one syllable), logographic (roughly, one character–one morpheme/word), and major encodings are addressed (in particular, Unicode). A description of the nature of speech and its representation in waveforms and spectrograms follows, including sections on how to read spectrograms and on language modeling using n-grams. The two latter sections come under the heading of "Under the Hood." Sections so headed (typically, one or two per chapter) provide digressions into selected technical material that provide bonuses for the interested reader, but which can be omitted without losing the gist of each chapter.

Chapter 2 brings up the first application, "Writer's Aids," which consists of two parts: checking spelling and checking grammar. The first part includes an overview of different kinds of spelling errors, and methods for detection and correction of errors. These include a description of dynamic programming for calculating the minimum edit distance between a misspelled word and a set of candidate corrections. The second part deals with grammar checking. To set the scene for this, there is first an exposition of context-free grammar for the purpose of specifying the norm (the well-formed sentences) of the language. (The possibility of enriching the formalism with features is mentioned later in the same chapter.) It is then stated that "[w]hen a parser fails to parse a sentence, we have an indication that something is grammatically wrong with the sentence" (page 58). At the same time, it is recognized that the most one can do is to build grammar fragments. So what about undergeneration, and the challenges of maintaining a large, manually encoded grammar? Instead of elaborating on this, the chapter goes on with a brief description of methods for correcting errors. These include relaxation-based techniques that discharge grammar rules, and special-purpose rules or n-grams that trigger directly on erroneous sequences of words. The reader is left wondering what one could expect from a grammar fragment. Although this may not be relevant to grammar checking, it would have been instructive to have a mention somewhere of the possibility of augmenting context-free rules or other formalisms with probabilities, and of how this has affected wide-coverage parsing (Clark and Curran 2007).

Chapter 3 deals with language tutoring systems and how to make them linguistically aware. To this end, the concepts (but not the inner workings) of tokenization, part-of-speech tagging, and syntactic parsing are described. An example language tutoring system for learners of Portuguese is then addressed. Compared with traditional workbook exercises, the system gives immediate feedback on orthographic, syntactic, and semantic errors, and also contains audio.

The topic of Chapter 4, "Searching," is mainly related to information retrieval. Most of the chapter is contained in two comprehensive sections that deal with searching in unstructured data (typically the Web) and semi-structured data (such as Wikipedia). The former section contains a relatively detailed description of search engines and PageRank, as well as an overview of HTML. Evaluation of search results is mentioned, but the measures are described in more detail in the next chapter. The section on semi-structured data contains a description of regular expressions, Unix *grep*, and finite-state automata. The chapter also contains brief sections on searching of structured data (databases, using Boolean expressions) and of text corpora (mainly discussing corpus annotation). Given that many of the readers of this book will be linguists, it is perhaps surprising that the book has so little material related to corpus linguistics, but it could be argued that this topic would require a text of its own (and that it is not an "application").

778

Anyway, the "Further Reading" section of the chapter includes some good suggestions on this topic.

Chapter 5 brings up document classification, and begins with an overview of concepts in machine learning. Then there is a detailed description of how to measure success in classification, with precision/recall, true/false positives/negatives, and the related measures of sensitivity and specificity used in medicine. After this, two examples of document classifiers are described in some detail: naive Bayes and the perceptron. Finally, there is a short section on sentiment analysis. This chapter has a good balance between applications-oriented and theoretical material, and gives a good grasp of each of them.

Chapter 6 deals with dialogue systems. After some motivation, the chapter examines in detail an example spoken dialogue transcript from the Carnegie Mellon University *Let's Go!* Bus Information System, followed by a thorough description of dialogue moves, speech acts, and Grice's conversational maxims. After this ambitious background, one would expect some (even superficial) material on the anatomy of a dialogue system founded on these principles, but instead there is a detailed description of *Eliza*. The motivation is that "[*Let's Go!*] is . . . too complicated to be explained fully in this textbook" (page 167). The ambition to explain a working system in full is admirable, but seems misdirected here. Why not try to work out the basic principles of a simpler question-answering system? As it stands, the applications-oriented material (*Eliza*) is not connected to the theoretical parts of the chapter. It is also potentially misleading when the authors say: "[*Eliza*] works reasonably well using simple means, and this can be useful if your application calls for a straightforward but limited way of creating the illusion of an intelligent being at the other end of the wire" (page 170). Here, the authors must be referring to chatbots, but for applications in general, it would have been valuable to have a pointer to principles of user interface design, such as trying to maintain consistency (Cohen, Giangola, and Balogh 2004). On a different note, this chapter would profit from picking up the thread on speech from Chapter 1. For example, it might be instructive to have an explanation of word-error rate and the degradation of input typically caused by a speech recognizer. This would also give an opportunity to elaborate on one of the problems mentioned at the outset of the chapter: "[f]ixing confusions and misunderstandings before they cause the conversation to break down" (page 154).

Chapter 7 brings up the final application, machine translation. This is a chapter that gives a good grasp of both applications-oriented and theoretical material. Example-based translation and translation memories are briefly discussed, and, after some background, word alignment, IBM Model 1, the noisy channel model, and phrase-based statistical translation are explained. Commercial translation is also addressed. In connection with the translation triangle, the possibility of an interlingua is discussed, the conclusion being that "[d]espite great efforts, nobody has ever managed to design or build a suitable interlingua" (page 189) and "we probably cannot have one any time soon" (page 190). This is true for a universal interlingua in the sense of a fully abstract language-independent representation, but what might be more relevant is domain-specific interlinguas, which have proved to be feasible in recent years (Ranta 2011). Interlinguas are also mentioned later in the chapter, at the end of "Under the Hood 12" (page 205) on phrase-based statistical translation, where it is stated that "[t]here is certainly no interlingua in sight." Although this is different, Google Translate actually uses English as an interlingua for a large majority of its language pairs (Ranta 2010). Surely, the reason for this is that Google typically has much more parallel data for language pairs where English is one of the members than for language pairs where this is not the case.

The final chapter ("Epilogue") brings up some philosophical questions: the impact of language technology on society (effects of automatization of natural-language tasks), the human communication system as opposed to animal communication, human self-perception when computers begin to use language to interact, and ethical considerations.

A good index is crucial in a book like this, also because it helps to fulfill the aim of showing the reader how the underlying techniques recur across applications. However, for numerous terms such as *n-gram, part-of-speech tagging, parsing, supervised learning,* and so forth, many (sometimes a majority) of the occurrences in the book are not covered by the index. It would be highly desirable to improve this for a future edition.

In sum, although a couple of chapters could have had a better balance or connection between the applications-oriented and theoretical material, the overall impression is that *Language and Computers* successfully fills a niche: It serves its purpose well in being an introductory textbook in computational linguistics for people without a computing background. It is possible to debate both the applications chosen and the theory selected for presentation, but it is nonetheless good to see a general introduction with a clear ambition to explain some of the inner workings of the technology.

I teach an introductory course in computational linguistics for Master's students in linguistics who lack a computing background. My experience is that the primary need of these students is corpus linguistics, because, above all, they need to learn how to practice linguistics as an empirical science. They soon reach a point, however, where they also need a broader introduction to the applications and methods of computational linguistics. Such a broad introduction is also needed by Bachelor's students in linguistics, and by those studying to become language consultants, translators, and so on. This book would then be the natural choice.

References

Clark, Stephen and James R. Curran. 2007. Wide-coverage efficient statistical parsing with CCG and log-linear models. *Computational Linguistics* 33(4):493–552.

Cohen, Michael H., James P. Giangola, and Jennifer Balogh. 2004. *Voice User Interface Design.* Addison-Wesley.

Jurafsky, Daniel and James H. Martin. 2009. *Speech and Language Processing: An Introduction to Natural Language Processing, Speech Recognition, and Computational Linguistics.* 2nd edition. Prentice-Hall.

Ranta, Aarne. 2010. Report from Google EMEA Faculty Summit 8–10 February 2010. http://www.molto-project .eu/node/842.

Ranta, Aarne. 2011. *Grammatical Framework: Programming with Multilingual Grammars.* CSLI Publications, Stanford, CA.

Mats Wirén is Associate Professor in computational linguistics at Stockholm University. He received his M.Sc. and Ph.D. from Linköping University. His current research focuses on language acquisition, whereas a long-time interest of his is parsing. He has also carried out research in speech translation and spoken dialogue. Wirén's address is Department of Linguistics, Stockholm University, SE-106 91 Stockholm, Sweden; e-mail: mats.wiren@ling.su.se.