Research Statement — Jason Eisner

Much of my research in computational linguistics has been bound up with two emerging strategies of the 1990's: statistical optimality and phonological optimality.

Statistics properly used can translate and enrich linguistic theories—rather than replacing them—and can connect those theories to processing models. That is, the statistical worldview need not be tied to simplistic engineering tricks: it is a plausible framework for rethinking linguistic competence, variation, and acquisition (see e.g. Abney 1996). Statistical techniques emerged in the computational community as a way of coping with real-world, ambiguity-riddled, sometimes ungrammatical language. I have worked to incorporate more **linguistic sophistication** into the statistical models—obtaining both linguistic and computational payoffs.

Optimality Theory, by contrast, has developed in the world of pure linguistics. It has been used primarily as a framework for thinking about phonologies, but has some computational trappings. I have worked to add more **computational sophistication** to OT—proposing constraints on constraints, a representational formalism, and an efficient generation algorithm that can handle infinite candidate sets. Again, the benefits are interdisciplinary: well-specified theories are more congenial for both linguists and computers.

Statistical Syntax

Statistics is a useful response to the poverty of the stimulus. It is the science of reasoning under uncertainty, for drawing conclusions from limited or incomplete or noisy data. And limited, incomplete, noisy data is exactly what we humans are faced with—not only when learning language, but also when conversing with anyone who prefers to mumble rather than draw full syntactic structures. I have been especially interested in statistical notions of *syntactic* competence and processing.

Parsing

A constraint on language comprehension is that it must take ambiguity in stride. A typical newspaper sentence has hundreds of possible parses. Humans unconsciously rule out almost all of these—avoiding implausible modifier attachments, and avoiding perverse conspiracies of rare grammatical constructions and word senses.

Since many parses are admissible, a good strategy is to bet on the most *probable* parse tree. In the case of a lexicalized grammar formalism (such as LFG, HPSG, LTAG, LCFG, CG, DG, or Minimalism), we can define a probable parse tree as one that is assembled from probable lexical entries.

Thus statistical parsing requires us to associate the entries of a syntactic lexicon with probabilities—yielding a kind of extended notion of syntactic competence. We need to know that "eat" is most commonly used as a simple transitive or intransitive verb, at 40% and 45% respectively. Less probable lexical entries for "eat" describe other possible local syntactic configurations, which may involve particles ("eat up"), passivization, object extraction, and so on.

In practice, this idea proves to be quite successful when combined with a statistical notion of selectional preferences, i.e., the obvious semantic notion that "eat" prefers certain complements and adjuncts to others. In 1992 at Bell Labs, Mark Jones and I built the first statistical parser to use such semantic preferences, which reduced our error rate fivefold. My subsequent work studied and improved the probability models and developed more efficient parsing algorithms, resulting in a 1996 parser that achieved a state-of-the-art 93% accuracy (with partial credit) on *Wall Street Journal* sentences.

Inducing Grammars From Examples (thesis)

Parsing with probabilities is all very well, but where does the grammar come from? We need to extrapolate a complete syntactic lexicon, with probabilities, based on only the entries that have shown up in a small collection of sample parses.

We could try to simply scan the sample parses and tally the various lexical entries used. Alas, there are never enough. For English we are fortunate to have a "treebank" of 50,000 hand-parsed *Wall Street Journal* sentences, created at a cost of $250,000—but for a typical new *Wall Street Journal* sentence, even the treebank omits 5–20% of the lexical entries needed for a correct parse.

For this reason, we must always flesh out the lexicon somehow. Current practice is to do a quick-and-dirty job, using crude statistical techniques. But is there a more principled and accurate approach?

The answer is yes. We need to discover and exploit the internal, language-specific structure of the lexicon.[1] The various lexical entries for "eat" mentioned earlier are not independent of one another. They are related by the syntactic principles of English, such as Bresnan's (1978) rule that derives passive lexical entries from active transitive ones. Knowing these principles is part of grammatical competence. If the sample parses use a rare word "snurf" as a transitive verb, a competent speaker can infer the corresponding passive lexical entry; or conversely, from the passive, she can reconstruct the active that it was probably derived from. Similarly, one can estimate the probability that "snurf" allows object drop, middle formation, heavy shift, and so on. The rules involved are essentially transformations that apply within the syntactic lexicon, where they can permute the local domain of a lexical head or introduce placeholders for long-distance gaps.

How to make use of this idea? There are several challenges. First, we need a way to correctly discover the language's lexical redundancy rules, such as active $\rightarrow$ passive, by studying the lexical entries that happen to be attested in the sample parses. Second, our rules have to attach probabilities to the new lexical entries they create. Third, in encoding systematic structure in the lexicon, we don't want to override the lexicon's traditional ability to represent well-attested idiosyncrasy: after all, some verbs don't passivize, and some passivize at an unusually high rate.

My approach is guided by Occam's Razor: other things equal, a lexicon is more likely to be right if it has more derived entries and fewer listed ones. In a probabilistic lexicon, a "derived" lexical entry is one whose associated probability is well-predicted from the other entries via lexical redundancy rules. Any deviation from this prediction corresponds to "listing" or "delisting" the entry, so we prefer deviations to be numerically close to zero.

The learner starts with a universal set of *possible* rules, which are initially assumed to apply infrequently. The rules compete in parallel, and apply in sequence, to convert lexical entries of various kinds into other lexical entries. An "obligatory transformation" is a rule that converts $A \rightarrow B$ at a rate near 100%; a rate of 50% can arrange for A and B to be about equally probable lexical entries, since half the A's are transformed; and a rate near 0% means that the transformation does not apply in the language.

The learner continuously adjusts the transformation rates and lexical entry probabilities in order to optimize the lexicon. On the one hand, we try to arrange for the lexical entries' probabilities to be well-predicted from each other via transformations, as discussed above. On the other, we also try to arrange for the probabilities to account well for the entries' observed frequencies among the sample parses. This delicate balance between generalization and empirical description is implemented by a hierarchical Bayesian model that also incorporates a number of other statistical techniques.

[1] Recall that in a lexicalized theory of grammar, the compositional apparatus is simple and universal. The lexicon is therefore the seat of all language-specific information—systematic as well as idiosyncratic.

In a direct comparison, this procedure induces better grammars from English newspaper text than any described in the literature. (The objective measure is cross-entropy, i.e., the ability of the induced lexicon to predict unseen test data.) Moreover, the transformations it identifies as having high rates in English can be inspected directly and are plausible. The approach is flexible enough to accomodate a wide range of linguistic frameworks, i.e., notions of what lexical entries and the rules transforming them should look like in natural language grammars.

Learning from Scratch (future work)

The Achilles' heel of the work described above is its dependence on a corpus of sample parses. One would like to reduce or eliminate that dependence. Children learn language from raw speech signals, situated in a real-world environment. Could a computer induce a grammar from raw text? What innate statistical biases would help it do so?

Raw text is intuitively useful in gradually *extending* one's grammatical competence. Humans can usually parse a sentence that contains a single novel lexical entry (i.e., a novel word or construction). Statistical parsers also have this ability, and for the same reason: the requirements of the surrounding context make a particular parse overwhelmingly likely.

Thus, a child or computer reading an age-appropriate text may have to skip the most difficult sentences, but there are other sentences—at the edge of its competence—for which it will posit novel lexical entries with high confidence. The strength of the transformation-learning model above is that it can generalize from these novel lexical entries, in a kind of one-shot learning. In short, parsing raw text can result in interesting new sample parses that can in turn be fed back into the statistical engine to improve the grammar.

I also plan to incorporate further linguistic ideas into the syntactic model. For example, the model currently cannot distinguish "NP ate" (object drop) from "NP opened" (decausative), although these distinctions in the θ-grid correlate usefully with selectional preferences. It also has no real ability to learn word classes (e.g., $\pm$ telic, $\pm$ animate).

Primitive Optimality Theory

Optimality Theory is an elegant framework for phonological description. In practice, though, it has produced something of a zoo of *ad hoc* descriptive constraints. To have a falsifiable theory making strong universal claims about the phonological apparatus, we must tackle some central questions:

- The **Con** question: *What constraints are allowed?*
- The **Gen** question: *What (kind of) representations do they constrain?*

Formalizing what OT can and cannot say is part of stating UG. In addition, computational work on OT is impossible without a (good or bad) formalization. I have proposed a restrictive formalization that is empirically motivated and easy to use, yet surprisingly clean and computationally tractable.

The Primitive Constraints

A reasonable start is to look at the hundreds of constraints that are proposed in the 200+ papers on the Rutgers Optimality Archive. They have more in common than one might think at first glance. Two simple, highly local families are ubiquitous, and can serve a unifying function:

(1) *Alignment/Licensing*: $\alpha \rightarrow \beta$ — "Each α overlaps with some β." (If not, it incurs one violation.)

(2) *Disalignment/Clash*: $\alpha \perp \beta$ — "Each α overlaps with no β." (One violation for each overlap.)

In these constraints, α and β may be either autosegmental objects (features, syllables, morphemes) or their edges.[2] I use a carefully formalized variant of autosegmental phonology in which autosegmental objects have explicit edges. Autosegmental association and I-O Correspondence are both represented simply by temporal overlap of the relevant objects along an unmarked "timeline."

The primitive constraint families are extremely well-attested. I have compiled a list of over 120 distinctive constraints from the OT literature, which shows each formal possibility allowed by these families to be widely used, not only cross-linguistically but across several domains of phonology: features, prosody, feature-prosody interaction, I-O relations (not limited to Correspondence), and morphophonology. A few examples are given in (3)–(6) at the end of this section.

Adequacy of the Primitive Constraints

What would happen if OT allowed *only* the primitive constraints? What would the resulting system of "primitive OT" (OTP) look like, and would it be descriptively adequate?

I have hardly been able to find any constraint in the literature that cannot be replaced by a primitive constraint or a small block of them. Relying on these carefully specified, well-attested families, and using their consistent notation, can make rigorous descriptive work in OT much easier to carry out. It seems that phonology may be mainly about local alignment and disalignment!

The one big exception is the Generalized Alignment (GA) family. The *primitive* alignment constraint given in (1) is purely local (cf. Zoll 1996). But GA needs to perform non-local, arithmetic tricks such as measuring the distance from every foot to the edge of the word and summing these distances. GA is a questionable mechanism for phonology, since it can be used to achieve unattested effects of greater than context-free power (!), such as requiring a pair of floating tones to fall $\frac{1}{4}$ and $\frac{3}{4}$ of the way through a word. OTP is provably unable to describe such a phonology; it is therefore more constrained than GA.

Can we eliminate GA in favor of the primitive constraints? The non-local, distance-counting features of GA have been most central in accounts of metrical stress. However, non-OT accounts of metrical stress have scrupulously avoided using any mechanisms with such power.

To answer this question, I have published an explanatory typology of metrical stress that uses only primitive constraints. "Iterative" footing effects result not from GA but from interactions of an extrametricality constraint with constraints on local stress clash or lapse. Indeed, the typology uses a single, coherent, rerankable set of *primitive* constraints to obtain *all* the basic facts about metrical stress[3]—including the well-known gaps and asymmetries in the non-OT paradigm of Hayes (1995). These gaps and asymmetries fall out of the account naturally; they can be traced back to the onset-coda asymmetry and the nearly universal preference for right-edge extrametricality. The typology also makes a new prediction—that left-to-right trochees should be incompatible with right-edge extrametricality—which is robustly confirmed in Hayes's survey of languages. GA-based accounts can stipulate these facts but explain none of them.

Computational Properties

On the simplest assumptions about Gen—roughly, that it can place anything anywhere—Optimality Theory has to choose among infinitely many candidates. This poses a challenge for processing. It also makes it difficult for linguists to be sure what their grammars predict.

[2]Certain limited forms of conjunction and disjunction can also be used when specifying α or β. For example, α may refer to voiced continuants.

[3]Iambic and trochaic foot form, quantity sensitivity, iambic lengthening, unbounded feet, simple word-initial and word-final stress, directionality of footing, syllable (and foot) extrametricality, degenerate feet, and word-level stress.

Once the system has been pinned down formally, however, we can bring computational techniques to bear on the generation problem. I have designed and implemented an efficient algorithm that can perform the required optimization for OTP grammars. The algorithm uses finite-state automata to represent the successively winnowed, potentially infinite candidate sets.

I have also proved a number of computational results about the formal power and computational complexity of OTP, and its relationship to other formal accounts of the phonological device.

Future Work

On the linguistic side, I am interested in continuing to study the descriptive and explanatory adequacy of the OTP formalism as new ideas develop in the OT community. One area for exploration concerns "sequential" versions of the primitive constraints—versions in which a violation incurred at point x is irredeemably worse than if it had been incurred at any point to the left of x. I would also like to explore the potential of OT syntax, although that area is still too immature to formalize.

On the computational side, the time is ripe for working on OTP algorithms that can do efficient comprehension and learning, not just generation.

Examples

(3) a. FILL-F(R): $]_F \rightarrow]_\sigma$ (cf. Inkelas, ROA-39)
"Feet must end on syllable boundaries."

b. NONFINALITY: $]_F \perp]_{PrWd}$ (Ní Chiosáin, ROA-89)
"Feet should not be word-final."

c. *ONS/N: ${}_\sigma[\perp {}_{nas}[$ (Smolensky, ROA-86)
"Syllables may not begin with nasal segments."

(4) a. NASVOI: $nas \rightarrow voi$ (Itô, Mester, & Padgett, ROA-38)
"Nasals must be voiced."

b. *TENSE-LOW: $tense \perp low$ (Benua, ROA-74)
"No tense low vowels."

c. PARSE μ: $\mu \rightarrow \sigma$ (Myers, ROA-6)
"Every mora must be contained in a syllable." (Actually "must overlap," but separate prosodic hierarchy constraints ${}_\sigma[\rightarrow {}_\mu[$ and $]_\sigma \rightarrow]_\mu$ prevent it from being only partly contained in the syllable.)

(5) a. MIN-2m: $F \rightarrow]_\mu$ (Green & Kenstowicz, ROA-101)
"A metrical foot contains at least two moras."

b. GEMINATE: $C \perp]_\sigma$ (Oostendorp, ROA-84)
"No geminate consonants."

(6) a. MAV(PRO): $round \rightarrow$ ($back$ or $stress$) (Féry, ROA-34)
"Umlauted vowels fall in prominent syllables. That is, a +round vowel that fails to be +back must bear stress."

b. CODACOND: ($]_C$ and $]_\sigma$) $\perp]_{lab}$ (cf. Lombardi, ROA-105)
"No syllable-final consonant may bear a labial feature."

Other Interests

I am interested in various formalisms for syntax, in semantic theory, and in the data that motivate these. I would also be happy to advise students interested in applied research in natural language engineering, such as text processing or information retrieval (IR), where I have co-authored patents.