Structure and Similarity

Tübingen July 3. 2002

Greg Kobele

1 Basic Ideas

The goals of this presentation are to

- provide a notion of structure for the purpose of comparing grammars that isn't affected by (natural) extensions of the grammar
  - if German has the same structure as Korean, then adding a simple NP (noun phrase) to German should not change this fact
- see how this notion of external structure can be used to capture internal regularities of the grammar

This is part of the bare grammar framework, which provides an 'unrestricted' framework for the description of language, the intention of which is to

- ensure that (what are proposed as) linguistic universals are actually properties of language, and not of the notation used to describe them (i.e. by making clear what are properties of the theory, and what are not)
- provide a framework into which other grammars can be translated

2 Mathematical Preliminaries

2.1 Functions

- a function $f : A \rightarrow B$ is
  - an **injection** iff $a \neq a'$ implies $f(a) \neq f(a')$, for all $a, a' \in A$
  - a **surjection** iff every $b \in B$ is $f(a)$ for some $a \in A$
  - a **bijection** iff it is both injective and surjective
- given two sets A and B, an **indexing** (or **naming**) of B by A is an onto function $f : A \rightarrow B$. For $b \in B$ and $a \in A$ such that $f(a) = b$, a is a name for b, written b_a. As every element has a name, $B = (b_a)_{a \in A}$.

2.2 Algebra

- a **signature** Σ is an I-indexed set of natural numbers, $\Sigma = (n_i)_{i \in I}$
- an **algebra** $\mathcal{A} = \langle A, F \rangle$, where
  - A is a set, called the **carrier** of the algebra
  - $F = (f_i)_{i \in I}$ is an I-indexed set of partial functions over A, such that $f_i : A^{n_i} \to A$, i.e. each f_i has arity n_i.

Example 1 Take $\mathcal{A} = \langle A, F \rangle$ such that

- $A = \mathbb{N}$,
- $F = \{+_0, 0_1\}$ (here $\Sigma = \{2_0, 0_1\}$, and $I = \{0, 1\}$)

then $\mathcal{A}$ is the monoid of natural numbers under addition.

2.2.1 Generators

Given an algebra $\mathcal{A} = \langle A, F \rangle$, $B \subseteq A$ is a set of **generators** for $\mathcal{A}$ iff the smallest set containing B that is closed under F is A.

- a set B is closed under an operation f just in case whenever $b_1, \ldots, b_n \in B$, and $f(b_1, \ldots, b_n)$ is defined, then $f(b_1, \ldots, b_n) \in B$.

2.2.2 Congruence Relations

An equivalence relation $\approx$ over the carrier set of an algebra $\mathcal{A}$ is a **(strong) congruence relation** if and only if, whenever

- $a_i \approx b_i$, and
- $f(a_1, \ldots, a_n)$ is defined, then

$f(a_1, \ldots, a_n) \approx f(b_1, \ldots, b_n)$

- Given $a \in A$, $a/\approx$ denotes the set of elements in A congruent to a: $a/\approx = \{a' | a \approx a'\}$. $a/\approx$ is called the 'block' containing a. The set $A/\approx = \{a/\approx | a \in A\}$ is a partition of A, and is said to induce $\approx$.

Example 2 The diagonal relation $\delta = \{\langle a, a \rangle | a \in A\}$ is a congruence relation in any algebra.

Example 3 In the monoid of natural numbers under addition, the partition $\pi = \{\{0\}, \{n|n > 0\}\}$ induces a congruence relation.

The congruence relations of an algebra form a lattice with δ as the bottom.

- meet is realized as intersection,
- join is the transitive closure of the set-theoretic union of the relations.

Example 4 in the monoid of natural numbers (as in any total algebra), the total relation $\omega = \{\langle a, a' \rangle | a, a' \in A\}$ is the top.

2.2.3 Quotient Algebra

Given a congruence $\approx$ of an algebra $\mathcal{A}$, the quotient algebra $\mathcal{A}/\approx = \langle A/\approx, F^{\approx}\rangle$ is defined by

- $A/\approx$ is the set $\{a/\approx |a \in A\}$
- if $f_i(a_1, \ldots, a_n) = a$, then $f_i^{\approx}(a_1/\approx, \ldots, a_n/\approx) = a/\approx$

2.2.4 Multi-arity Algebra

- a **multi-arity signature** Σ is an I-indexed subset of $2^{\mathbb{N}}$
  $\Sigma = (\phi_i)_{i \in I}$, for $\phi_i \subseteq \mathbb{N}$
- a **(partial) multi-arity algebra** is a tuple $\mathcal{A} = \langle A, F \rangle$, where:
  - A is the carrier of the algebra,
  - $F = (f_i)_{i \in I}$ is an I-indexed set, where
    * each f_i is a partial operation over A, such that f_i is the union of partial functions $f_{i,n} : A^n \to A$, for each $n \in \phi_i$

3 Bare Grammar

The Bare Grammar framework is designed to be as unrestrictive as possible

- we want universal properties of grammars to be linguistic, not notational universals

- we want to be able to translate different formalisms into bare grammars

We view a language as being built from a set of lexical items, by a set of structure building functions, that build increasingly complex expressions from simpler ones.

- a bare grammar consists of the language so built, and a set of structure building functions. In other words, a bare grammar is a multi-arity partial algebra $G = \langle L, F \rangle$.

Example 5 The grammar below (**Little Korean** from (Keenan & Stabler, 97)), illustrates this formalism. It is intended to capture the anaphor-antecedent relation of (actual) Korean, whereby a reflexive pronoun ('himself') can precede its antecedent, as long as the reflexive is accusative, and the antecedent is nominative.
The lexicon consists of:
NP: John, Bill, himself
K: -nom, -acc
C: and
P1n: laughed
P2: praised
The rules are CM (case mark), PA (predicate argument), and COORD (coordination), where:

Domain	CM	Value	Condition
s, NP + -nom, K	→	s⌢-nom, KPn	s ≠ himself
s, NP + -acc, K	→	s⌢-acc, KPa	none

Domain	PA	Value	Condition
s, KPx + t, P1x	→	s⌢t, S	x ∈ {n, a}
s, KPx + t, P2	→	s⌢t, P1y	x ≠ y, and x, y ∈ {n, a}

Domain	COORD	Value	Condition
and, C + s, X + t, X	→	both⌢s⌢and⌢t, X	X ∉ {C, NP}

Letting L denote the smallest set containing the lexicon that is closed under CM, PA and COORD, **Little Korean** $= \langle L, \{CM, PA, COORD\} \rangle$.

3.1 Structure

As a bare grammar is essentially a set with structure (in the form of the (aptly named) structure building operations), a simple definition of (internal) structure suggests itself: we look at those properties/relations which do not change under 'structure preserving' maps over the language L.

Intuitively, a map is structure preserving if it does not change how expressions are built up. Formally, a structure preserving map is an automorphism π of the language L. An automorphism π is an isomorphism from the algebra to itself. An isomorphism is:

- a bijection $\pi : L \to L$, such that
- $\pi(f_i(a_1, \ldots, a_n)) = f_i(\pi(a_1), \ldots, \pi(a_n))$

A **structural** property/relation P is one which is fixed by every automorphism π, i.e. $\pi(P) = P$.

Example 6 In **Little Korean**,

- each case marker (-nom, K and -acc, K) is structural,
- for each X, the phrases of category X are structural,
- the co-argument relation, given by s CO-ARGUMENT t IN u iff for some v of category P2, either PA(s, PA(t, v)) or PA(t, PA(s, v)) are constituents (to be defined) of u
- the possible antecedant relation is structural, where s is a possible antecedent of t in u iff s is a co-argument of t in u and t is a phrase of category KPa

Example 7 In every grammar,

- the constituent of relation is structural:

  $sCON_0 s$

  $sCON_{n+1}t$ iff there are some $u_1, \ldots, u_i, \ldots, u_n \in L$ and some $f_i \in F$ such that $t = f_i(u_1, \ldots, u_i, \ldots, u_n)$ and $sCON_n u_i$

  $sCONt$ iff $sCON_n t$ for some n

- the greatest congruence relation (i.e. the top element in the lattice) is not necessarily structural

4 Structure across Bare Grammars

4.1 Automorphisms

The set of automorphisms of a bare grammar has structure: it is closed under composition, taking inverses, and it has an identity element (it is a group). A natural way to compare the structure of two grammars would be to compare the automorphism groups of each. Two grammars would have the same structure iff their automorphism groups were isomorphic.

4.1.1 Problems

This solution doesn't capture our intuitions about similarity

- not restrictive enough

  Example 8 Let $G = \langle\{a, b\}, \{\}\rangle$. Then the automorphism group of G is isomorphic to the automorphism group of **Little Korean**.

- Also, too restrictive.

  Example 9 Let G' be the grammar obtained by adding the NP 'Sam' to **Little Korean** and extending the structure building functions to treat Sam similarly to Bill (and closing the extended language under the new functions). Then the automorphism groups of **Little Korean** and of G' are not isomorphic.

4.2 A Notion of Behaviour

Why are we reluctant to say that G and **Little Korean** don't have the same structure after all? The answer pursued here is that the expressions of G behave completely differently from the expressions of **Little Korean**.

Proposal: Two expressions have the same behaviour iff they are syntactically indistinguishable.

Syntactic indistinguishability seems to work as follows. Two items are syntactically indistinguishable iff:

- they have the same distribuition - the domains of the structure building operations are closed under arbitrary replacement of one for the other
- whether or not such substitution has taken place is not syntactically determinable - the result of such a substitution is syntactically indistinguishable from the original

Certainly any expression is syntactically indistinguishable from itself. Moreover, if a is syntactically indistinguishable from b, b should be syntactically indistinguishable from a, and if a and b, and b and c are syntactically indistinguishable, then so are a and c.

These intuitions motivate the following definition:

An equivalence relation $\approx$ is a syntactic indistinguishability relation iff

- if $x \approx y$ and $f_i(a_1, \ldots, x, \ldots, a_n)$ is defined, then $f_i(a_1, \ldots, x, \ldots, a_n) \approx f_i(a_1, \ldots, y, \ldots, a_n)$

Theorem 1 $\approx$ is a syntactic indistinguishability relation iff it is a congruence relation.

Given Theorem 1 it makes sense to look at the quotient of a bare grammar under a syntactic indistinguishability relation. For purposes of comparing structure, we are interested only in the top element of the congruence lattice - i.e. that congruence that identifies two elements whenever it is possible to do so. We can then apply a new notion of structure across grammars:

> $G = \langle L, F\rangle$ has the same behaviour as $G' = \langle L', F'\rangle$ iff $G/\approx$ is isomorphic to $G'/\approx'$, where $\approx, \approx'$ are the greatest congruences of G and G' respectively.

We can see that if G has the same behaviour as G', then extensions of G with expressions that behave similarly to existing expressions also have the same behaviour as G' (and thus also as G).

5 More Structure in a Bare Grammar

5.1 Problems with Automorphisms

Example 10 The following grammar (**Little Spanish** from (Keenan & Stabler, 97)) is designed to capture certain agreement facts in languages like spanish, where adjectives and determiners agree in gender with nouns.
The lexicon consists of:
Nm: man, doctor
Nf: woman, obstetrician
A: gentle, intelligent
D: every, some
Agr: -a, -o
Amod: very, moderately

The generating functions are: GM (gender marking), PA (predicate argument), where

Domain	GM	Value	Condition
s, X + -o, Agr	$\rightarrow$	s$\frown$-o, Xm	X $\in$ {D, A}
s, X + -a, Agr	$\rightarrow$	s$\frown$-a, Xf	X $\in$ {D, A}

Domain	PA	Value	Condition
s, Dy + t, Ny	$\rightarrow$	s$\frown$t, NPy	y $\in$ {m, f}
s, Ay + t, Ny	$\rightarrow$	s$\frown$t, Ny	y $\in$ {m, f}
s, Amod + t, Ay	$\rightarrow$	s$\frown$t, Ay	y $\in$ {m, f}

Letting L be the smallest set containing the lexicon which is closed under GM and PA, **Little Spanish** $= \langle L, \{GM, PA\}\rangle$.

5.2 Stablility

In **Little Spanish**, gentle and intelligent, and -a and -o are interchangeable by some automorphism. Yet gentle and intelligent seem to have more in common with one another than do -a and -o.

One might try to formalize this intuition in a number of ways:

- there are more automorphisms that interchange gentle and intelligent than those that interchange -a and -o (degree of relatedness depends on the number of automorphisms that interchange them)

- there are automorphisms that interchange gentle and intelligent that permute fewer expressions than any automorphism that interchanges -a and -o (degree of relatedness depends on the number of other elements one needs to permute in order to interchange them)

- the fact that there exist any automorphisms that interchange -a and -o depends upon there being an equal number of masculine and feminine nouns - a seemingly accidental property of the grammar. We might want to restrict our attention to automorphisms that are 'stable' in the sense that they do not depend on equinumerosity to interchange elements (degree of relatedness depends on the kind of automorphism that permutes them)

The first two ideas seem to depend on the number of other elements in the grammar, while we would like a way to express the relatedness of gentle and intelligent that doesn't depend on such 'extraneous' considerations. The final suggestion is more difficult to evaluate, and has been proposed in (Keenan & Stabler, 97).

We have to place restrictions on the types of expressions we can 'add' to our grammar when evaluating the stability of automorphisms (if we want to have the stable automorphisms interchange gentle and intelligent but not -o and -a)

- the added expressions must behave similarly to the already existing expressions

  Example 11 Let G' be the result of adding some expression X on which the generating functions are defined as follows: PA(X, gentle) = gentle. Then in G' there is an automorphism that interchanges -a and -o but no automorphism interchanges gentle and intelligent.

- the added expressions must be 'new', in the sense that they cannot be constituents of any already existing expression

Example 12 (**Stabler**) Let G' be the result of adding the expression 'very intelligent' of category A to **Little Spanish**. Then no automorphism interchanges intelligent and gentle in G'.

5.3 Categories and Behaviour

Notice that, while the proposal of stability (when restricted as per the above) works to distinguish the relatedness of -a and -o from the relatedness of gentle and intelligent, it relies on the notion of behaviour to do this; there is a much simpler way to capture this similarity using strictly less formal machinery: gentle $\approx$ intelligent, but -a $\not\approx$ -o, where $\approx$ is the maximal congruence relation over **Little Spanish**.

While the assignment of categories to expressions in the examples have been somewhat arbitrary (and, indeed, completely unnecessary considering our definition of a grammar), it is noteworthy that they line up rather closely with the equivalence classes of the maximal congruences. This is exemplified in formalisms such as categorial grammar, in which the syntactic category of an expression (usually) directly reflects its distribution/syntactic behaviour.

We might dispense altogether with categories in light of the above comments, and rely instead exclusively on the equivalence classes of the syntactic indistinguishability relation.

5.4 Subcategories, Co-occurrence Restrictions and Higher order Regularities

Though this notion of behaviour provides a neat way to compare the structure of two grammars, and also a foundation on which to define a category, there are many things that it does not suffice to do.

Subcategories: There are distributional similarities that this fine-grained notion of behaviour misses - for example, the similarity between himself and Bill in **Little Korean**.

- one might formulate this relationship as one of having certain 'features' in common, or as having the same category, but being in a different subcategory (e.g. NP, and reflexive NP)

- one possibility is to define 'congruences' which are indexed to individual functions, such that

  if $x \approx^{f_i} y$ then if $f_i(a_1, \ldots, x, \ldots, a_n)$ is defined, then $f_i(a_1, \ldots, x, \ldots, a_n) \approx^{f_i} f_i(a_1, \ldots, y, \ldots, a_n)$

Co-occurrence Restrictions and Higher order Regularities: a higher order regularity[1] is one which goes 'beyond' mere surface distributional similarity, and requires some sort of abstraction - an example is principle C in english[2]:

1. John loves Bill.
2. *John loves John.
3. Bill loves John.
4. *Bill loves Bill.

Here, John and Bill have a different distribution on the surface, but this 'different' surface distribution can be unified if we allow ourselves an abstraction: "*'X loves X', X an NP"

5.5 Wedding Automorphisms and Behaviour

Recall in §5.2 I offered three notions of 'relatedness' of expressions (repeated here):

- there are more automorphisms that interchange gentle and intelligent than those that interchange -a and -o (degree of relatedness depends on the number of automorphisms that interchange them)
- there are automorphisms that interchange gentle and intelligent that permute fewer expressions than any automorphism that interchanges -a and -o (degree of relatedness depends on the number of other elements one needs to permute in order to interchange them)
- the fact that there exist any automorphisms that interchange -a and -o depends upon there being an equal number of masculine and feminine

[1] I don't quite have a grip on what a 'higher order regularity' is...

[2] I think that this is not a syntactic phenomenon, but a semantic one. It is however illustrative of a higher order regularity.

nouns - a seemingly accidental property of the grammar. We might want to restrict our attention to automorphisms that are 'stable' in the sense that they do not depend on equinumerosity to interchange elements (degree of relatedness depends on the kind of automorphism that permutes them)

I dismissed the first two as depending on the number of other expressions in the language. If we look at the quotient algebra, these notions are invariant under 'addition' of elements to the grammar.

- Perhaps this is a way to deal with higher order regularities???

6 Behaviourally Optimal Grammars

Our notions of structure (automorphisms, congruences) are closely tied up with our particular choice of generating functions. *On a purely syntactic level* we can ask whether we have the 'best' functions for this purpose...

6.1 Partitioning the Class of Universal Algebras by Weak Isomorphism

A 'weak' isomorphism doesn't care about the identities of the functions relating tuples to expressions, just about whether each tuple - expression relation that obtains in one algebra, obtains in the other:

> a weak isomorphism is a bijection $\pi : \langle A, (f_i)_{i \in I} \rangle \rightarrow \langle B, (g_j)_{j \in J} \rangle$ such that
>
> - if $f_i(a_1, \ldots, a_n) = a$, then there is some $j \in J$ such that $g_j(h(a_1), \ldots, h(a_n)) = h(a)$
> - if $g_j(h(a_1), \ldots, h(a_n)) = h(a)$, then there is some $i \in I$ such that $f_i(a_1, \ldots, a_n) = a$