1 General introduction

1. Chomsky (1957): a **language** is a set of its grammatical sentences.

1.1. Change “sentences” to “expressions”.

1.2. Change “expressions” to “expressions with their structural descriptions”.

1.2.1. String similarity is usually misleading:

(1) a. John is easy to please.
b. John is eager to please.

(2) a. I expected John to leave.
b. I persuaded John to leave.

(3) a. We expected several students to be at the talk.
b. We persuaded several students to be at the talk.

(4) a. A unicorn seems to be in the garden.
b. A unicorn tries to be in the garden.

(5) a. It is easy to play **this sonata** on **this violin**.
b. **This sonata** is easy to play on **this violin**.
c. **This violin** is easy to play **this sonata** on.

(6) a. John grows tomatoes.
b. John destroys tomatoes.

1.3. Item 1.2. brings meaning into picture:

> It is assumed in *LSLT* (as in *SS*) that the theory developed is to be embedded in a broader semiotic theory which will make use of the structure of *L*, as here defined, to determine the meaning and reference of expressions and the conditions on their appropriate use, and will also encompass other investigations (statistical linguistics, etc.). (Chomsky 1975:3)

2. **Grammar** is an explicit system of rules and representations that pairs phonetic forms (sound pathway) with logical forms (meaning pathway).

3. **Acceptability** (data) versus **grammaticality** (theory).

3.1. Factors effective in acceptability are manifold and complex.

3.2. Grammar is just one among many.

3.3. Such idealization is indispensable in science.

3.3.1. Economist Dani Rodrik:

> All models are wrong. They are helpful [when] used in relevant context. Empirics without models yield no understanding.
>
> To clarify, models are wrong in the same sense that a subway map is wrong. Leaves out, misrepresents real world details.
>
> Simplicity in theory is a feature, not a bug. “But the real world is more complicated” is never good riposte. All causal theories simplify. (tweets, March 8, 2017)

4. **Descriptive** versus **explanatory** adequacy.

4.1. Description: “Given a language, what is possible to utter to mean what?”

4.1.1. A descriptively successful grammar for a given language has the widest possible coverage with a minimal set of rules and assumptions.

4.2. Explanation: “What is a possible human language?”

4.2.1. What is the common denominator of all the descriptively successful grammars?

2 Phrase Structure

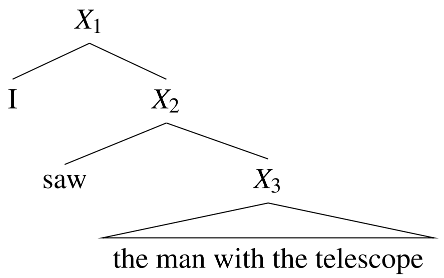
1. “Grammar is about things that go together in meaning” (Mark Steedman)

(7) a. ((Toy car) manufacturer.)
b. (Toy (car manufacturer.))

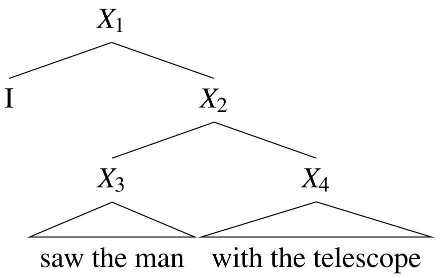
2. A motivation for phrase structure: structural ambiguity.

(8) I saw the man with the telescope.

(9)



(10)



3. A motivation for phrase structure + categories: distributional generalizations.

3.1. The labels X_2 and X_3 are identical when it comes to what can come in the context *I saw* ___.

3.2. Likewise, *I* and *saw* are instances of larger classes that can appear in the places they occupy.

4. Another motivation for phrase structure comes from the need for relational generalizations:

(11) a. John murdered Smith.
b. Smith was murdered by John.

(12) a. The army destroyed the city.
b. The army's destruction of the city.
c. The city's destruction by the army.

(13) a. We expected several students to be at the talk.
b. We expected there to be several students at the talk.

(14) a. John saw the cat that chased the dog.
b. Which dog did John see the cat that chased?

(15) a. John saw the man with the telescope.
b. Which telescope did John see the man with?

(16) a. The man_i who said he_i was tall.
b. *The man_i who he_i said was tall.

5. Such generalizations are best stated over phrase structures.

Complex NP Constraint (Ross 1986):

(17) a. I believed the claim that Otto was wearing this hat.
b. I believed that Otto was wearing this hat.

(18) a. *Which hat did you believe the claim that Otto was wearing?
b. Which hat did you believe that Otto was wearing?

Complex Subject Constraint (Ross 1986):

(19) a. The reporters expected [that the principal would fire some teacher].
b. [That the principal would fire some teacher] was expected.
c. It was expected by the reporters [that the principal would fire some teacher].

Noun phrases in *that*-clauses can be relativized in (19a) and (19c), but not in (19b) (Ross 1986:147).

3 Categories and Features

1. Features start to appear in linguistic generalizations originally in phonetics.

1.1. Some simplified phonetics:
Features: (i) where (**place**) in the vocal tract it is articulated (alveolar, dental, glottal, and so on), (ii) how (**manner**) it is articulated (fricative, plosive,

nasal, and so on); and (iii) whether the vocal chords are involved or not (**voice**). A particular consonant can be represented as a set of features.

For example, *t* and *d* are both "alveolar plosives", articulated by restricting (hence plosive) the air flow by tongue and alveolar ridge (hence alveolar). The former is voiceless, while the latter is voiced. Formally, *t* is {[place alveolar],[manner plosive],[voice -]}, while *d* is {[place alveolar],[manner plosive],[voice +]}. We adopt a abbreviating convention, where we omit the words "place" and "manner" and write '[+voice]' instead of '[voice +]', etc.

In its full complexity, the consonant /g/ would be just a short-hand for (Gazdar *et al.* 1985:17):

(20) $$\begin{bmatrix} +\ \text{SEGMENT} \\ +\ \text{CONSONANTAL} \\ -\ \text{SONORANT} \\ -\ \text{SYLLABIC} \\ +\ \text{HIGH} \\ +\ \text{BACK} \\ -\ \text{LOW} \\ -\ \text{ROUND} \\ -\ \text{ANTERIOR} \\ -\ \text{CORONAL} \\ -\ \text{NASAL} \\ -\ \text{CONTINUANT} \end{bmatrix}$$

2. Chomsky (1970:207): "[t]he distinction between features and categories is a rather artificial one."

   Chomsky (1970:208): "we might just as well eliminate the distinction of feature and category, and regard all symbols of the grammar as sets of features."

3. Features become useful in various levels of specification: V is enough for tense, but you need subcat for transitive versus intransitive modes of combination.

4. Initially 3 important types of features: projection level (word versus phrase), head features (agreement), combinatory features (valency for now, directionality and order later).

3.1 Feature structures

Definition 3.1 (Feature structures without variables[1])
Let S_c be the set of symbols standing for complex features and S_a for atomic features. The set of feature structures F is the smallest set satisfying,

i. If $\alpha \in S_c$ and $\beta \in S_a$, then $\{\langle \alpha, \beta \rangle\} \in F$.

ii. If $\alpha \in S_c$ and $\beta \in F$, then $\{\langle \alpha, \beta \rangle\} \in F$.

iii. If F_1 and $F_2 \in F$, and $\mathrm{Dom}(F_1) \cap \mathrm{Dom}(F_2) = \emptyset$, then $F_1 \cup F_2 \in F$.

□

1. Attribute-value matrices (AVMs) make things more readable.

1.1. Represent $\{\langle \alpha, \beta \rangle\}$ as $\begin{bmatrix} \alpha & \beta \end{bmatrix}$.

1.2. Represent $\{\langle \alpha_1, \beta_1 \rangle, \ldots, \langle \alpha_n, \beta_n \rangle\}$ as $\begin{bmatrix} \alpha_1 & \beta_1 \\ \vdots & \vdots \\ \alpha_n & \beta_n \end{bmatrix}$.

1.3. Here is a definition of feature structures with variables, in AVM notation:

Definition 3.2
Let S_c be the set of symbols standing for complex features and S_a for atomic features. The set of feature structures F is the smallest set satisfying,

i. If $\alpha \in S_c$ and $\beta \in S_a$, then $\begin{bmatrix} \alpha & \beta \end{bmatrix} \in F$.

ii. If $\alpha \in S_c$ and $\beta \in S_a$ and $n \in \mathbb{Z}^+$, then $\begin{bmatrix} \alpha & \boxed{n}\beta \end{bmatrix} \in F$.

iii. If $\alpha \in S_c$ and $n \in \mathbb{Z}^+$, then $\begin{bmatrix} \alpha & \boxed{n} \end{bmatrix} \in F$.

[1] See the appendix for a set-theoretic formulation of feature structures with variables.

iv. If $\alpha \in S_c$ and $\beta \in F$, then $\begin{bmatrix}\alpha & \beta\end{bmatrix} \in F$.

v. If $\gamma \in F$ and not formed by this rule, and $n \in \mathbb{Z}^+$, then $\boxed{n}\,\gamma \in F$.

vi. If $F_1 = \begin{bmatrix}\alpha_1 & \beta_1\\ \vdots & \vdots\\ \alpha_m & \beta_m\end{bmatrix}$ and $F_2 = \begin{bmatrix}\gamma_1 & \delta_1\\ \vdots & \vdots\\ \gamma_n & \delta_n\end{bmatrix}$ for $m,n \geq 1$ are both $\in F$,
and if, disregarding any possible indices heading α_i and γ_j,

$$\{\alpha_1,\ldots,\alpha_m\} \cap \{\gamma_1,\ldots,\gamma_n\} = \emptyset, \text{ then } \begin{bmatrix}\alpha_1 & \beta_1\\ \vdots & \vdots\\ \alpha_m & \beta_m\\ \gamma_1 & \delta_1\\ \vdots & \vdots\\ \gamma_n & \delta_n\end{bmatrix} \in F.$$

□

2. Sag *et al.*'s (2003) differ from Definition 3.2 by having a type decoration for each feature structure. Type decorations can be fitted into the system of Definition 3.2 by introducing an obligatory feature TYPE for each feature structure. The following equivalence would hold:

(21)
$$\begin{bmatrix}\textit{phrase} & \\ \text{HEAD} & \begin{bmatrix}\textit{verb} & \\ \text{AGR} & \begin{bmatrix}\text{NUM} & \text{sg}\\ \text{PER} & \text{3rd}\end{bmatrix}\end{bmatrix}\\ \text{VAL} & \begin{bmatrix}\textit{val-cat} & \\ \text{COMPS} & \text{itr}\\ \text{SPR} & +\end{bmatrix}\end{bmatrix} \equiv \begin{bmatrix}\text{TYPE} & \text{phrase}\\ \text{HEAD} & \begin{bmatrix}\text{TYPE} & \text{verb}\\ \text{AGR} & \begin{bmatrix}\text{NUM} & \text{sg}\\ \text{PER} & \text{3rd}\end{bmatrix}\end{bmatrix}\\ \text{VAL} & \begin{bmatrix}\text{TYPE} & \text{val-cat}\\ \text{COMPS} & \text{itr}\\ \text{SPR} & +\end{bmatrix}\end{bmatrix}$$

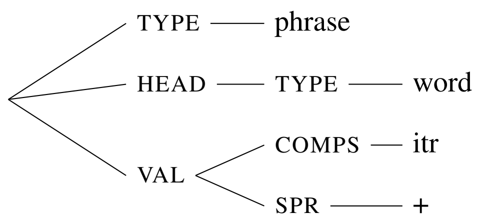
3. Sag *et al.*'s (2003) type decorations should not be confused with atomic features; their notation is a little confusing in this. For example, the following sentence category,

(22)
$$\begin{bmatrix}\textit{phrase} & \\ \text{HEAD} & \textit{verb}\\ \text{VAL} & \begin{bmatrix}\text{COMPS} & \text{itr}\\ \text{SPR} & +\end{bmatrix}\end{bmatrix}$$

is actually,

(23)
$$\begin{bmatrix}\textit{phrase} & \\ \text{HEAD} & \begin{bmatrix}\textit{verb}\end{bmatrix}\\ \text{VAL} & \begin{bmatrix}\text{COMPS} & \text{itr}\\ \text{SPR} & +\end{bmatrix}\end{bmatrix} \equiv \begin{bmatrix}\text{TYPE} & \text{phrase}\\ \text{HEAD} & \begin{bmatrix}\text{TYPE} & \text{verb}\end{bmatrix}\\ \text{VAL} & \begin{bmatrix}\text{COMPS} & \text{itr}\\ \text{SPR} & +\end{bmatrix}\end{bmatrix}$$

4. Feature structures can also be represented as **directed acyclic graphs** (DAGs) with a unique root constraint. For example the AVM in (23) is,

(24)


5. A path in such a DAG is a successive application of feature names to a feature structure to access its sub-features. For instance, naming the above AVM F, we have,

$$F(<\text{TYPE}>) = \text{phrase}$$
$$F(<\text{VAL COMPS}>) = \text{itr}$$

3.2 Subsumption and Unification

1. A feature structure F_1 **subsumes** another F_2, shown $F_1 \sqsubseteq F_2$ if F_2 has all the information F_1 has.

   Definition 3.3 (Subsumption)
   Given F_1 and F_2:

   (a) $F_1 \sqsubseteq F_2$, if F_1 and F_2 are atomic features and equal.

   (b) $F_1 \sqsubseteq F_2$, if F_1 and F_2 are feature structures such that
   $\mathrm{Dom}(F_1) \subseteq \mathrm{Dom}(F_2)$ and
   for each $f \in \mathrm{Dom}(F_1)$, $F_1(\langle f \rangle) \sqsubseteq F_2(\langle f \rangle)$.

   □

2. Given two feature structures F_1 and F_2, their **unification** $F_3 = F_1 \sqcup F_2$, if exists, is the smallest feature structure subsumed by both, that is $F_1 \sqsubseteq F_3$ and $F_2 \sqsubseteq F_3$.

3. Shieber 1986 is a classic on the use of unification in grammar.

3.3 Structure sharing (re-entrancy)

1. The variables in AVMs indicate token rather than type identity.

2. It might be tempting to think that the AVMs in (25) and (26) are equivalent, in the sense that (25) is just a more economic way of representing (26); but they are not.

$$(25)\quad \begin{bmatrix} \text{HEAD} & \boxed{1}\begin{bmatrix} \text{AGR} & \begin{bmatrix} \text{PER} & \text{2nd} \end{bmatrix} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SPEC} & \begin{bmatrix} \text{HEAD} & \boxed{1} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

$$(26)\quad \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{AGR} & \begin{bmatrix} \text{PER} & \text{2nd} \end{bmatrix} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SPEC} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{AGR} & \begin{bmatrix} \text{PER} & \text{2nd} \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

2.1. In (26) there is an accidental type identity between certain parts of the structure; while (25) states that certain sub-structures of the AVM are token identical – they are one and the same thing.

2.2. The difference is best understood by observing their behavior under unification. Take the following AVM for instance,

$$(27)\quad \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{AGR} & \begin{bmatrix} \text{NUM} & \text{pl} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

While unifying (27) with (26) gives,

$$(28)\quad \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{AGR} & \begin{bmatrix} \text{PER} & \text{2nd} \\ \text{NUM} & \text{pl} \end{bmatrix} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SPEC} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{AGR} & \begin{bmatrix} \text{PER} & \text{2nd} \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

unifying it with (25) gives,

$$(29)\quad \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{AGR} & \begin{bmatrix} \text{PER} & \text{2nd} \\ \text{NUM} & \text{pl} \end{bmatrix} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SPEC} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{AGR} & \begin{bmatrix} \text{PER} & \text{2nd} \\ \text{NUM} & \text{pl} \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

4 Heads, complements, specifiers

1. Endocentric and exocentric constructions.

2. The notion of head:

2.1. most “important” item in the phrase;

2.2. same distribution with the entire phrase – possible, not necessary;

2.3. obligatory in their phrase;

2.4. imposes restrictions on its dependents – their number and form.

3. Typical head-dependent relations:

3.1. part-of-speech restriction;

3.2. agreement;

3.3. case

4.1 Adjunct/complement

1. How to tell:

1.1. If adjunct, then optional; equivalently, if obligatory, then complement.

1.2. The form of a complement is governed by its head; adjuncts have no syntactic or semantic restrictions – only pragmatic plausibility.

1.3. Adjuncts can be indefinitely iterated in any orders; complements are limited in number and come in more restricted orders – case take over for order in some languages.

2. Adjunct/complement distinction is cross-categorial; e.g. complement adverbials, adjunct NPs.

3. All objects are complements, but not vice-versa.

3.1. “Object” usually means a complement with an accusative case (direct object) or dative case (indirect object). In,

(30) John resorted to violence.

to violence is a prepositional complement of the verb *resort*; *violence* is an object of the preposition *to*. The complement *to violence* is obligatory (**John resorted*), the form of the complement is governed by *resort* (**John resorted in violence*.

3.2. Primary test for objecthood is passivization:

(31) a. I gave John the book.
b. John was given the book by me.
c. *The book was given John.

(32) a. I gave the book to John.
b. The book was given to John by me.
c. *To John was given the book.
d. *John was given the book to.

(33) a. John resorted to violence.
b. *To violence was resorted by John.
c. Violence was resorted to by John.[2]

3.3. Some non-objective complements of verbal heads:

(34) a. John relies [on Sue]. (prepositional)
b. John wants [to leave]. (infinitival)
c. John thinks [that he is a genius]. (finite/clausal)
d. Sue considers Bill [intelligent]. (adjectival)
e. Sue considers [leaving Bill]. (gerundive)
f. Sue treated Bill [cruelly]. (adverbial)

3.4. See Tallerman 2015:123–4 for typical complements of non-verbal heads.

4. Transitive/intransitive distinction concerns whether a head gets an objective complement or not. Terminology becomes loose when it comes to “ditransitives”.

5. The term “argument” usually appears in its semantic sense – as components (=relata) of the relation denoted by the head.

[2] On the basis of passivization data, it has been claimed that the preposition *to* is actually the part of the verb rather than the complement TODO:citation.

5 Formalizing Subcategorization

1. The aim of this section is to better understand in what way using feature structures and unification helps in building “better” grammars.

2. The motivation behind having a class for verbs is the existence of some grammatical properties that apply to every member of this class. For instance, tense is a grammatical category that is present in verbs but not in other classes like nouns, adjectives, prepositions, etc.

3. Although verbs are alike in this respect, they widely differ in the number and the way they come together with their arguments. Remember that “argument” is a semantic term, related to the conceptual categories involved in the event, state or action denoted by the verb. Syntax is the level where verbs come together with expressions that denote their arguments.

3.1. For an example take the verb *frighten*. At the conceptual level *frighten* denotes a psychological relation with two essential components: a stimulus and an experiencer. The verb *frighten* is a particular way of “syntacticizing” this relation in English. The syntax of this verb is such that the linguistic expression that denotes the experiencer argument is placed to the right of the verb and receives the accusative form (*him*, *her*) when it is denoted by a pronoun. Another way to syntacticize the same relation, again in English, is the verb *fear*. This verb reverses the syntactic coding: experiencer becomes the leftward nominative noun phrase and the stimulus becomes the rightward accusative noun phrase.

3.2. The concept of fear has other realizations in English. In cases where the stimulus is unknown or unimportant English speakers have recourse to the adjectival predicate *be scared*, as in *John is scared*. For such cases **John fears* does not work.

3.3. On the other hand, no syntacticization of the concept of fear makes the manner or means of creating fear a syntactic complement; this component of the concept is left to the syntactic process of adjunction, as in *Bill frightened Sue by his malicious grin.*

3.4. For another example take the verb *make*, as in *Sue makes Bill mad*. The verb expresses the relation of inflicting a certain mental quality on someone. It is the syntax of this verb that decides the quality (=being mad) should come after the patient (=Bill).[3]

3.5. To sum up, verbs are linguistic manifestations of relational concepts and their syntactic properties determine which arguments are realized in what form, order and direction.

3.6. “Subcategorization” is a technical term that might sometimes be misleading. It is related to subcategories of the class “verb” formed according to combinatory properties – number, form, order, directionality of complements. But it is usually used as a verb itself, as in “The verb *hit* subcategorizes for a direct object,” and “the verb *make* subcategorizes for a direct object and an adjectival phrase.”

4. We now turn to formalizing subcategorization. Let us not worry about specifiers for the moment and concentrate on complements.

5. As a first shot, we can have:

$$\begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{CAT} & \text{v} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{COMPS} & \text{itr} \end{bmatrix} \end{bmatrix} \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{CAT} & \text{v} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{COMPS} & \text{tr} \end{bmatrix} \end{bmatrix} \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{CAT} & \text{v} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{COMPS} & \text{dtr} \end{bmatrix} \end{bmatrix}$$

for intransitive (*sleep*), transitive (*hit*) and ditransitive (*give*) verbs.

This would require a separate PS (=phrase structure) rule for each type of verb. For instance, the following is for transitives:[4]

$$\begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{CAT} & \text{v} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{COMPS} & \text{itr} \end{bmatrix} \end{bmatrix} \rightarrow \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{CAT} & \text{v} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{COMPS} & \text{tr} \end{bmatrix} \end{bmatrix} \ \text{NP}$$

[3] English is not absolutely strict in this order. When the direct object is a “heavy” noun phrase the order is swapped: *Sue made mad, the guy who had been running after her for the last seven years* with an intonational break indicated by the comma. The phenomenon is called Heavy NP Shift.

[4] NP is a shorthand for $\begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{CAT} & \text{n} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SPEC} & + \end{bmatrix} \end{bmatrix}$.

6. We will need more rules to be able to capture the variety of verbs like those in (34), and many more.

7. Apart from the issue of the proliferation of rules, there is a problem of redundancy. The information that *want* can go with an infinitival complement (e.g. *to leave*) will be stated twice: one in the lexical entry of this verb – by a feature – and the other in the rule that combines *want* with its complement.

8. One solution would be to list *want* as:

$$\begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{CAT} & \text{v} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{COMPS} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{CAT} & \text{v} \end{bmatrix} \\ \text{VFORM} & \text{to-inf} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

and have the rule for one complement taking verbs as:

$$\begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{CAT} & \text{v} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{COMPS} & \text{nil} \end{bmatrix} \end{bmatrix} \rightarrow \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{CAT} & \text{v} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{COMPS} & \boxed{1} \end{bmatrix} \end{bmatrix} \boxed{1}$$

9. This will enable us to form *hit the ball* and *want to leave* with the same rule – of course we need a lexical entry for *hit* that lists its complement as an NP.

10. We need more categories for *want*, to be able to capture:

(35) a. John wants an ice-cream.
b. John wants Sue to stay.
c. John wants Bill arrested.
d. John wants Bill on his knees.

6 An example

1. Start with proper names, not worrying about semantics, and having minimal information in the beginning. All we care is that *ali* is a noun rather than a verb or a postposition.[5]

(36) $$\begin{bmatrix} \text{PHON} & \text{/ali/} \\ \text{SYN} & \begin{bmatrix} \text{CAT} & \text{n} \end{bmatrix} \end{bmatrix} \begin{bmatrix} \text{PHON} & \text{/ayşe/} \\ \text{SYN} & \begin{bmatrix} \text{CAT} & \text{n} \end{bmatrix} \end{bmatrix}$$

2. Let us assume the Turkish lexicon has the entries:

(37) $$\begin{bmatrix} \text{PHON} & \text{/uyuyor/} \\ \text{SYN} & \begin{bmatrix} \text{CAT} & \text{v} \end{bmatrix} \end{bmatrix} \begin{bmatrix} \text{PHON} & \text{/andırıyor/} \\ \text{SYN} & \begin{bmatrix} \text{CAT} & \text{v} \end{bmatrix} \end{bmatrix}$$

3. We need rules to have our lexical entries add up to sentences:

(38) $$\begin{bmatrix} \text{PHON} & \boxed{1}+\boxed{2} \\ \text{SYN} & \begin{bmatrix} \text{CAT} & \text{s} \end{bmatrix} \end{bmatrix} \rightarrow \begin{bmatrix} \text{PHON} & \boxed{1} \\ \text{SYN} & \begin{bmatrix} \text{CAT} & \text{n} \end{bmatrix} \end{bmatrix} \begin{bmatrix} \text{PHON} & \boxed{2} \\ \text{SYN} & \begin{bmatrix} \text{CAT} & \text{v} \end{bmatrix} \end{bmatrix}$$

(39) $$\begin{bmatrix} \text{PHON} & \boxed{1}+\boxed{2}+\boxed{3} \\ \text{SYN} & \begin{bmatrix} \text{CAT} & \text{s} \end{bmatrix} \end{bmatrix} \rightarrow \begin{bmatrix} \text{PHON} & \boxed{1} \\ \text{SYN} & \begin{bmatrix} \text{CAT} & \text{n} \end{bmatrix} \end{bmatrix} \begin{bmatrix} \text{PHON} & \boxed{2} \\ \text{SYN} & \begin{bmatrix} \text{CAT} & \text{n} \end{bmatrix} \end{bmatrix} \begin{bmatrix} \text{PHON} & \boxed{3} \\ \text{SYN} & \begin{bmatrix} \text{CAT} & \text{v} \end{bmatrix} \end{bmatrix}$$

4. How do these entries and rules work together? Sometimes it is easier to think of these matters in procedural terms.

5. Starting from the bottom, you may take the surface string as given, and ask whether this string constitutes a well-formed expression of the language. Here is an informal algorithmic recipe.

5.1. Take the input as a list of expressions.

[5] The items that are called "prepositions" in English have correlates in Turkish that come after, rather than before their complements: *masanın üstü*. When the order is not specified, the name of the category is "adposition".

5.2. For each expression in the list, pick an entry from the lexicon, whose phon feature matches the phonetic form of the input. If there is more than one entry for a given item, you randomly pick one of them.

5.3. Replace items in your list with their lexical entries. Now you have a list of lexical entries, rather than just items.

5.4. Randomly pick a rule. If the rule has *n* items on the right hand side, see whether your first *n* items fits these slots.

5.4.1. If they fit, unify them with the slots and replace those *n* items with the left hand side of the rule. Now you have an altered list. Go back to 5.4. with this altered list.

5.4.2. If they don't fit, repeat step 5.4..

5.5. If you get convinced that no rule among your rules can fit the first *n* items in your list; try step 5.4. with a list that leaves out the first element. But if you succeed with this list in step 5.4.1. and will need to go back to 5.4., put back the item you set aside at the beginning of the list.

5.6. Try to reduce your list to a single item in as many ways possible. After exhausting all the possible rule and item combinations, go back to step 5.2. and try other lexical selections.

6. An example: analysing *ali uyuyor*,

6.1. you get the following list of entries:

(40) $$\left\langle \begin{bmatrix}\text{PHON} & \text{/ali/}\\ \text{SYN} & \begin{bmatrix}\text{CAT} & \text{n}\end{bmatrix}\end{bmatrix}, \begin{bmatrix}\text{PHON} & \text{/uyuyor/}\\ \text{SYN} & \begin{bmatrix}\text{CAT} & \text{v}\end{bmatrix}\end{bmatrix} \right\rangle$$

6.2. If you hit upon 38, you can combine the two elements in your list into one, and stop there. Combination means unification of categories. You start from the left hand side, and from left to right – actually the order does not matter. Unifying the entry for *ali* with the first category on the left hand side of 38 will instantiate the phon value to '/ali/'. Likewise, the second category gets unified with the second item in the list. The instance of the rule gets turned into,

(41) $$\begin{bmatrix}\text{PHON} & \text{/ali/+/uyuyor/}\\ \text{SYN} & \begin{bmatrix}\text{CAT} & \text{s}\end{bmatrix}\end{bmatrix} \rightarrow \begin{bmatrix}\text{PHON} & \text{/ali/}\\ \text{SYN} & \begin{bmatrix}\text{CAT} & \text{n}\end{bmatrix}\end{bmatrix} \begin{bmatrix}\text{PHON} & \text{/uyuyor/}\\ \text{SYN} & \begin{bmatrix}\text{CAT} & \text{v}\end{bmatrix}\end{bmatrix}$$

6.3. Therefore, the two items in the list get replaced with one on the right hand side of 41. Trying other combinations of lexical choices and rule will not yield any further results for this particular case, so we stop.

7. A similar treatment of *ali andırıyor*, *ayşe ali andırıyor* would also yield successful derivations of a sentence; showing that our grammar is not sound – it treats some non-sentences as sentences.

8. The reason for this unsoundness (or overgeneration) is that neither our lexical entries nor rules are capable of distinguishing between *uyuyor* and *andırıyor*. As a first step to a remedy, we list *uyuyor* as,

(42) $$\begin{bmatrix}\text{PHON} & \text{/uyuyor/}\\ \text{SYN} & \begin{bmatrix}\text{CAT} & \text{v}\\ \text{COMPS} & \begin{bmatrix}\text{SYN} & \begin{bmatrix}\text{CAT} & \text{n}\end{bmatrix}\end{bmatrix}\end{bmatrix}\end{bmatrix}$$

and revise Rule 38 as:[6]

(43) $$\begin{bmatrix}\text{PHON} & \boxed{1}+\boxed{2}\\ \text{SYN} & \begin{bmatrix}\text{CAT} & \text{s}\end{bmatrix}\end{bmatrix} \rightarrow \begin{bmatrix}\text{PHON} & \boxed{1}\\ \text{SYN} & \boxed{4}\end{bmatrix} \begin{bmatrix}\text{PHON} & \boxed{2}\\ \text{SYN} & \begin{bmatrix}\text{CAT} & \text{v}\\ \text{COMPS} & \begin{bmatrix}\text{SYN} & \boxed{4}\end{bmatrix}\end{bmatrix}\end{bmatrix}$$

9. We still can derive *ali uyuyor*. What we avoid to derive are non-sentences like *ali ayşe uyuyor*, which were possible to get derived previously. It is crucial, however, to observe that these modifications do not help with avoiding *ali andırıyor*, since there is nothing to prevent the lexical entry of *andırıyor* to unify with the second category on the right hand side of 43. For this to

[6]Note that we are treating the subject as a complement of the verb, contrary to what we have been seeing so far. We do so, because we decided to start with minimal assumptions, and nothing so far suggest that we should not treat subjects as complements of verbs.

be prevented, we need to specify the comps feature of *andırıyor* such that it becomes non-unifiable with the category aimed for *uyuyor* type intransitive verbs. But how can we specify the comps feature of a verb like *andırıyor*, which takes two complements. Sag *et al.* (2003) does it with allowing feature values that are lists. We will do it in a slightly different way, but after we attend to some other inadequacies of our grammar.

10. Our grammar does not handle agreement. Introducing lexical items for other persons and numbers will result in further overgeneration, for instance *siz uyuyor*. The worst way of handling agreement is to populate our lexicon with forms like *uyuyorum*, *uyuyoruz* etc. Missing the systematic nature of verbal inflection would be costly. Assuming verbs have 4 rounds of inflection on average with 5 possible values for each round would multiply the lexical entries by 625 – and this is only for verbs.

11. What is the minimal agreeing form in Turkish verbs? It is not *uyor*, *uyorum*, *uyorsun*, etc. but "no marking", *um*, *sun*, for third , first and second persons, respectively. We better list these items in the lexicon, so that we can use them in a systematic way. We also need an entry for the progressive suffix *uyor*. We need to make sure that (i) progressive comes before agreement, (ii) we have exactly one instance of each, not more, not less. Let us make them categories on their own, so that we can regulate matters by rules.

(44)

$$\begin{bmatrix} \text{PHON} & \text{/uyu/} \\ \text{SYN} & \begin{bmatrix} \text{CAT} & \text{v} \\ \text{COMPS} & \begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{CAT} & \text{n} \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

$$\begin{bmatrix} \text{PHON} & \text{/iyor/} \\ \text{SYN} & \begin{bmatrix} \text{CAT} & \text{t} \\ \text{COMPS} & \begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{CAT} & \text{v} \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

$$\begin{bmatrix} \text{PHON} & \text{/um/} \\ \text{SYN} & \begin{bmatrix} \text{CAT} & \text{ag} \\ \text{AGR} & \text{1sg} \\ \text{COMPS} & \begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{CAT} & \text{t} \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

$$\begin{bmatrix} \text{PHON} & \text{/sun/} \\ \text{SYN} & \begin{bmatrix} \text{CAT} & \text{ag} \\ \text{AGR} & \text{2sg} \\ \text{COMPS} & \begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{CAT} & \text{t} \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

$$\begin{bmatrix} \text{PHON} & \text{//} \\ \text{SYN} & \begin{bmatrix} \text{CAT} & \text{ag} \\ \text{AGR} & \text{3sg} \\ \text{COMPS} & \begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{CAT} & \text{t} \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

12. These entries say that the complement of tense categories is v, and the complement of agreement category is t. Therefore, t category should turn v into t, so that agreement can combine with it and a further t cannot combine with it. Here are the rules:

(45)

$$\begin{bmatrix} \text{PHON} & \boxed{1}+\boxed{2} \\ \text{SYN} & \begin{bmatrix} \text{CAT} & \text{t} \end{bmatrix} \end{bmatrix} \rightarrow \begin{bmatrix} \text{PHON} & \boxed{1} \\ \text{SYN} & \boxed{4} \end{bmatrix} \begin{bmatrix} \text{PHON} & \boxed{2} \\ \text{SYN} & \begin{bmatrix} \text{CAT} & \text{t} \\ \text{COMPS} & \begin{bmatrix} \text{SYN} & \boxed{4} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

(46)

$$\begin{bmatrix} \text{PHON} & \boxed{1}+\boxed{2} \\ \text{SYN} & \begin{bmatrix} \text{CAT} & \text{ag} \\ \text{AGR} & \boxed{5} \end{bmatrix} \end{bmatrix} \rightarrow \begin{bmatrix} \text{PHON} & \boxed{1} \\ \text{SYN} & \boxed{4} \end{bmatrix} \begin{bmatrix} \text{PHON} & \boxed{2} \\ \text{SYN} & \begin{bmatrix} \text{CAT} & \text{ag} \\ \text{AGR} & \boxed{5} \\ \text{COMPS} & \begin{bmatrix} \text{SYN} & \boxed{4} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

13. These will derive forms like *uyuyor*, *uyuyorum*, *uyuyorsun* as ag type categories with a specific agreement information. To avoid non-sentences like *siz uyuyor*, we need our n categories to be specified for agreement features:

(47) $$\begin{bmatrix}\text{PHON} & /\text{ali}/ \\ \text{SYN} & \begin{bmatrix}\text{CAT} & \text{n}\\ \text{AGR} & \text{3sg}\end{bmatrix}\end{bmatrix} \quad \begin{bmatrix}\text{PHON} & /\text{ben}/ \\ \text{SYN} & \begin{bmatrix}\text{CAT} & \text{n}\\ \text{AGR} & \text{1sg}\end{bmatrix}\end{bmatrix} \quad \begin{bmatrix}\text{PHON} & /\text{sen}/ \\ \text{SYN} & \begin{bmatrix}\text{CAT} & \text{n}\\ \text{AGR} & \text{1sg}\end{bmatrix}\end{bmatrix}$$

14. Now we make our sentence forming rule sensitive to agreement:

(48) $$\begin{bmatrix}\text{PHON} & \boxed{1}+\boxed{2} \\ \text{SYN} & \begin{bmatrix}\text{CAT} & \text{s}\end{bmatrix}\end{bmatrix} \rightarrow \begin{bmatrix}\text{PHON} & \boxed{1} \\ \text{SYN} & \boxed{4}\end{bmatrix} \begin{bmatrix}\text{PHON} & \boxed{2} \\ \text{SYN} & \begin{bmatrix}\text{CAT} & \text{ag} \\ \text{AGR} & \boxed{5} \\ \text{COMPS} & \begin{bmatrix}\text{SYN} & \boxed{4}\begin{bmatrix}\text{CAT} & \text{n}\\ \text{AGR} & \boxed{5}\end{bmatrix}\end{bmatrix}\end{bmatrix}\end{bmatrix}$$

15. The current grammar achieves what we wanted. It cannot derive things like *ben uyuyor* and can derive grammatical sentences. However, it looks strange that it is the sentence rule that tells that there is one n complement to the ag head. This does not cause any problems for the moment, only because we left out *andır* out of the picture, to come back to it later. The problem is this: it is the verb that decides how many complements to take, but we are giving that discretion to the sentence forming rule. We need to correct this. The way to correct this is to keep the subcategorization information of the verb "alive" through all combinations with tense and agreement. We revise the rules for combining t and ag as follows:

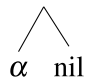
(49) $$\begin{bmatrix}\text{PHON} & \boxed{1}+\boxed{2} \\ \text{SYN} & \begin{bmatrix}\text{CAT} & \text{t} \\ \text{COMPS} & \boxed{6}\end{bmatrix}\end{bmatrix} \rightarrow \begin{bmatrix}\text{PHON} & \boxed{1} \\ \text{SYN} & \boxed{4}\begin{bmatrix}\text{COMPS} & \boxed{6}\end{bmatrix}\end{bmatrix} \begin{bmatrix}\text{PHON} & \boxed{2} \\ \text{SYN} & \begin{bmatrix}\text{CAT} & \text{t} \\ \text{COMPS} & \begin{bmatrix}\text{SYN} & \boxed{4}\end{bmatrix}\end{bmatrix}\end{bmatrix}$$

(50) $$\begin{bmatrix}\text{PHON} & \boxed{1}+\boxed{2} \\ \text{SYN} & \begin{bmatrix}\text{CAT} & \text{ag} \\ \text{AGR} & \boxed{5} \\ \text{COMPS} & \boxed{7}\end{bmatrix}\end{bmatrix} \rightarrow \begin{bmatrix}\text{PHON} & \boxed{1} \\ \text{SYN} & \boxed{4}\begin{bmatrix}\text{COMPS} & \boxed{7}\end{bmatrix}\end{bmatrix} \begin{bmatrix}\text{PHON} & \boxed{2} \\ \text{SYN} & \begin{bmatrix}\text{CAT} & \text{ag} \\ \text{AGR} & \boxed{5} \\ \text{COMPS} & \begin{bmatrix}\text{SYN} & \boxed{4}\end{bmatrix}\end{bmatrix}\end{bmatrix}$$

With these modifications, the tense and agreement attaching rules project the complement information of the verb.

16. So far we try to model intransitive verbs like *uyu*. Now let us move on to transitives like *andır*. **?** make use of lists in modelling multiple complement taking.[7] We will follow a similar strategy, differing in the structure we use to represent lists. We will treat lists as binary branching trees, where the left branch holds the initial element of the list and the right node holds the rest of the list, itself as a list with the same binary branching tree structure. The special symbol nil will stand for the empty list. In this scheme, a single element list like $\langle\alpha\rangle$ will be represented as:

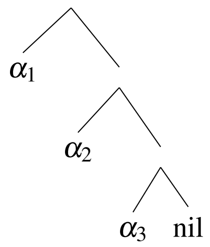
(51)



A three element list $\langle\alpha_1, \alpha_2, \alpha_3\rangle$ will become:

[7]Note that in their exposition a verb like *andır* takes only one complement and one specifier, the subject. Currently we do not make a specifier/complement distinction in verbs; therefore, for us *andır* is a two-complement verb.

(52)



17. List structures will be represented in attribute-value matrices as follows:

(53) $$\begin{bmatrix} \text{FIRST} & \alpha \\ \text{REST} & \text{nil} \end{bmatrix}$$

(54) $$\begin{bmatrix} \text{FIRST} & \alpha_1 \\ \text{REST} & \begin{bmatrix} \text{FIRST} & \alpha_2 \\ \text{REST} & \begin{bmatrix} \text{FIRST} & \alpha_3 \\ \text{REST} & \text{nil} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

18. We now adjust our two verbs:

(55) $$\begin{bmatrix} \text{PHON} & \text{/uyu/} \\ \text{SYN} & \begin{bmatrix} \text{CAT} & \text{v} \\ \text{COMPS} & \begin{bmatrix} \text{FIRST} & \begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{CAT} & \text{n} \end{bmatrix} \end{bmatrix} \\ \text{REST} & \text{nil} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

$$\begin{bmatrix} \text{PHON} & \text{/andır/} \\ \text{SYN} & \begin{bmatrix} \text{CAT} & \text{v} \\ \text{COMPS} & \begin{bmatrix} \text{FIRST} & \begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{CAT} & \text{n} \end{bmatrix} \end{bmatrix} \\ \text{REST} & \begin{bmatrix} \text{FIRST} & \begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{CAT} & \text{n} \end{bmatrix} \end{bmatrix} \\ \text{REST} & \text{nil} \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

19. Let us now, for a moment, forget about tense and agreement, and concentrate on what kind of rules can we devise for feeding in the complements of a verb.

References

Chomsky, N. (1957). *Syntactic Structures*. Mouton, The Hague.

Chomsky, N. (1970). Remarks on nominalization. In R. Jacobs and P. Rosenbaum, editors, *Readings in English Transformational Grammar*, pages 184–221. Ginn, Waltham, Mass.

Chomsky, N. (1975). *The Logical Structure of Linguistic Theory*. Plenum Press, New York.

Gazdar, G., Klein, E., Pullum, G. K., and Sag, I. A. (1985). *Generalized Phrase Structure Grammar*. Harvard University Press, Cambridge, MA.

Ross, J. R. (1986). *Infinite Syntax!* Ablex Publishing Corporation, Norwood, NJ.

Sag, I. A., Wasow, T., and Bender, E. M. (2003). *Syntactic Theory: A Formal Introduction*. CSLI, Stanford, CA.

Shieber, S. (1986). *An Introduction to Unification-Based Approaches to Grammar*. CSLI Publications, Stanford, CA.

Tallerman, M. (2015). *Understanding Syntax*. Hodder Education, Croydon, UK, 4 edition.

A Definitions

Definition A.1 (Feature structures with variables – variable clash possible)
Let S_c be the set of symbols standing for complex features, S_a for atomic features, both disjoint from $\mathbb{Z}^+$. Let K_c be the set of indexed complex feature names defined as $\{\langle n, \alpha\rangle \mid \alpha \in S_c \text{ and } n \in \mathbb{Z}^+\}$, and likewise let K_a be the set of indexed atomic feature names defined as $\{\langle n, \alpha\rangle \mid \alpha \in S_a \text{ and } n \in \mathbb{Z}^+\}$.

The set of feature structures F is the smallest set satisfying,

1. If $\alpha \in K_c$ and $\beta \in K_a \cup \mathbb{Z}^+$, then $\{\langle \alpha, \beta\rangle\} \in F$.
2. If $\alpha \in K_c$ and $\beta \in F$, then $\{\langle \alpha, \beta\rangle\} \in F$.
3. If F_1 and $F_2 \in F$, and $\text{Range}(\text{Dom}(F_1)) \cap \text{Range}(\text{Dom}(F_2)) = \emptyset$, then $F_1 \cup F_2 \in F$.

□