Formal Models of Language: Formal versus Natural Language

Paula Buttery

Easter 2018

Formal vs. Natural Languages

We can define a **formal language** precisely as a set of strings over an alphabet (see the *Grammars* handout), but what is the definition of a **natural language**? A natural language can be thought of as a mutually understandable communication system that is used between members of some population. When communicating, speakers of a natural language are tacitly agreeing on what strings are allowed (i.e. which strings are ***grammatical***?[1]). Dialects and specialised languages (including e.g. the language used on social media) are all natural languages in their own right. Note that named languages that you are familiar with, such as *French*, *Chinese*, *English* etc, are usually historically, politically or geographically derived labels for populations of speakers rather than linguistic ones.[2]

what is a natural language?

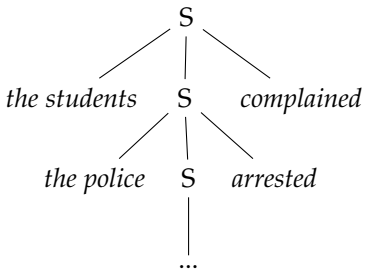
[1] Grammaticality has traditionally been considered a binary property of any given string – the string is either grammatical or it is not – however, recent work has shown that grammaticality can be gradient, with some strings found to be 'more' grammatical than others, based on native speakers' judgements (see for example `dx.doi.org/10.1111/cogs.12414`).

[2] Chinese for instance encompasses both Cantonese and Mandarin which are not mutually intelligible languages; and some of the Scandinavian languages, each of which have their own name (*Swedish*, *Danish*) might better be thought of as mutually intelligible dialects. There are various dialect continua, for example between German and Dutch, whereby geographically-juxtaposed dialects are mutually intelligible, but dialects at either 'end' of the continuum (e.g. central German and south-eastern Dutch) are not.

1. Language Complexity

In the *Grammars* handout we noted a trade-off between the ***expressivity*** of a language class and the algorithmic ***running time*** for recognising a string from a language in that class. An important question then is whether all natural languages can be modelled using the class of regular grammars. This is an important question for two reasons: first, it places an upper bound on the running time of algorithms that process natural language; second, it may tell us something about human language processing and language acquisition (more on this in later sections). It turns out that regular grammars have limitations when modelling natural languages for several reasons:

Centre Embedding In principle, the syntax of natural languages cannot be described by a regular language due to the presence of centre-embedding; *i.e.* infinitely recursive structures described by the rule, $A \rightarrow \alpha A \beta$, which generate language examples of the form, $a^n b^n$. For instance, the sentences below have a centre-embedded structure.



1. The students the police arrested complained.
2. The luggage that the passengers checked arrived.
3. The luggage that the passengers that the storm delayed checked arrived.[3]

Intuitively, the reason that a regular language cannot describe centre-embedding is that its associated automaton has no memory of what has occurred previously in a string. In order to 'know' that n verbs were required to match n nominals already seen, an automaton would need to 'record' that n nominals had been seen; but a DFA has no mechanism to do this. A formal proof uses the ***pumping lemma*** property to show that strings of the form $a^n b^n$ are not regular.[4] Careful here though: a regular grammar could generate constructions of the form $a^* b^*$ but not the more exclusive subset containing only $a^n b^n$ (which would represent centre embeddings). More generally the complexity of a sub-language is not necessarily the complexity of a language. If we show that the English subset of string of the form $a^n b^n$ is not regular it does ***not*** follow that English itself is not regular. To prove something about the complexity of English, we can use the knowledge that regular languages are closed under intersection. So if we assume English is regular and intersect it with another regular language (e.g. the one generated by $/the\ a\ (that\ the\ a)^* b^*/$) we should get another regular language. However the intersection of a regular language of form $a^* b^*$ with English results in constructions of the form $a^n b^n$ (in our example case $/the\ a\ (that\ the\ a)^{n-1} b^n/$), which is not regular as it fails the pumping lemma property. The assumption that English is regular must be wrong.

However, examples of centre-embedding quickly become unwieldy for human processing (*n.b.* the difficulty of understanding the example sentences above). For finite n we can still model the language using a DFA/regular grammar: we can design the states to capture finite levels of embedding. So are there any other reasons not to use regular grammars for modelling natural language?

Redundancy Grammars written using regular grammar rules alone are highly redundant: since the rules are very simple we need a great many of them to describe the language. This makes regular grammars very difficult to build and maintain.

Useful internal structures There are instances where a regular language[5] can recognise the strings of a language but in doing so

[3] Regular languages are closed under ***homomorphism***: this means we can map all the *nouns* to a and all the *verbs* to b and then describe centre embeddings in 2. and 3. to be of the general form $/the\ a\ (that\ the\ a)^{n-1} b^n/$.

[4] For each $l \geq 1$, find some $w \in \mathcal{L}$ of length $\geq l$ so that no matter how w is split into three, $w = u_1 v u_2$, with $|u_1 v| \leq l$ and $|v| \geq 1$, there is some $n \geq 0$ for which $u_1 v^n u_2$ is not in $\mathcal{L}$. To prove that $\mathcal{L} = \{a^n b^n | n \geq 0\}$ is not regular. For each $l \geq 1$, consider $w = a^l b^l \in \mathcal{L}$.

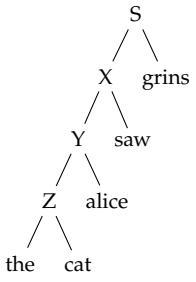
If $w = u_1 v u_2$ with $|u_1 v| \leq l$ & $|v| \geq 1$, then for some r and s:

- $u_1 = a^r$
- $v = a^s$, with $r + s \leq l$ and $s \geq 1$
- $u_2 = a^{l-r-s} b^l$

so $u_1 v^0 u_2 = a^r \epsilon a^{l-r-s} b^l = a^{l-s} b^l$

But $a^{l-s} b^l \notin \mathcal{L}$ so by the Pumping Lemma, $\mathcal{L}$ is not a regular language

[5] Below: A left-branching tree structure derivable from some RG (ie. all rules of form $A \to Bb$ for $A, B \in \mathcal{N}$ and $b \in \Sigma$). This structure does not capture linguistic constituency.



does not provide a structure that is linguistically useful to us. The left-linear or right-linear internal structures derived by regular grammars are generally not very useful for higher level NLP applications. We need informative internal structure so that we can, for example, build up good semantic representations.[6]

In practice, regular grammars can be useful for ***partial grammars*** (i.e. when we don't need to know the syntax tree for the whole sentence but rather just some part of it) and also when we don't care about derivational structure (i.e. when we just want a Boolean for whether a string is in a language). For example, in information extraction, we need to recognise NAMED ENTITIES. These are essentially referents *e.g.* The Computer Lab, Prof. Sir Maurice Wilkes, the Backs, Great Saint Mary's, the Gog Magog Hills, and so on. The internal structure of named entities is normally unimportant to us, we just want to recognise when we encounter them.

For instance, using rules such as:

$$\begin{aligned} NP &\rightarrow nnsb\ NP \\ NP &\rightarrow np1\ NP \\ NP &\rightarrow np1 \end{aligned}$$

where NP is a non-terminal and $nnsb$ and $np1$ are terminals representing tags from the large CLAWS2 166 tag set,[7] you could match a titled name like, *Prof. Stephen William Hawking*.[8]

So the next question is whether the class of context-free grammars is expressive enough to model natural language. Or in other words, for every natural language that exists, can we find a context-free grammar to generate it?

There is some evidence that natural language can contain CROSS-SERIAL DEPENDENCIES. A small number of languages exhibit strings of the form shown in Figure 1.

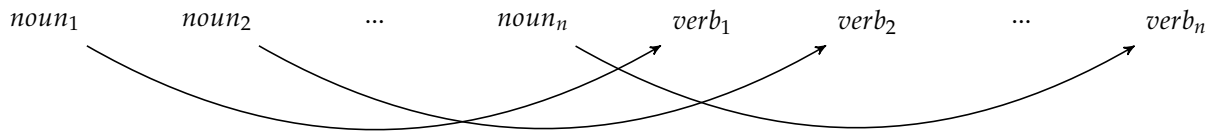


Figure 1: A schematic for cross-serial dependencies in language.

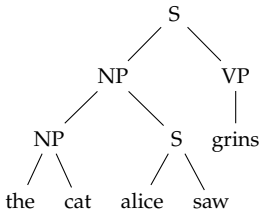
There is a Zurich dialect of Swiss German in which constructions like the following are found:

mer d'chind em Hans es huus haend wele laa hälfe aastriiche.
we the children Hans the house have wanted to let help paint.
we have wanted to let the children help Hans paint the house

Such expressions may not be derivable by a context-free grammar.[9]

If we are to use formal grammars to represent natural language, it is useful to know where they appear in the hierarchy (especially since the decision problem is intractable for languages above

[6] Below: a tree structure that captures linguistic constituency derived from a CFG (ie. all rules of form $A \rightarrow \alpha$ where $A \in \mathcal{N}$ and $\alpha \in (\Sigma \cup \mathcal{N})^*$). Note that NP and VP are single non-terminal symbols not two in a row—in linguistic terminology they represent a ***noun phrase*** (a phrased *headed* by a noun) and a ***verb phrase*** respectively.



[7] You can find the CLAWS2 tag set at `http://ucrel.lancs.ac.uk/claws2tags.html`. *nnsb* tags a preceding noun of style or title, abbr. (such as *Rt.* or *Hon.*); and *np1* tags singular proper nouns (such as *London*, *Jane* or *Frederick*).

[8] Note that although noun phrases can be structurally complicated (*e.g.* the man who likes the dog which bites postmen), the relative clause is not generally part of a named entity so we don't need to capture it in the grammar (i.e. we use a partial grammar).

[9] The proof follows similarly as that for centre embeddings except that we must use the pumping lemma for context-free languages.

context-free in the hierarchy). However, notice that we can in fact divide the space of all languages any way we see fit; we are not limited to discussing language classes only in terms of the Chomsky hierarchy.

With respect to natural language, it might turn out that the set of all attested natural languages is actually as depicted in Figure 2: note the overlap with the context-sensitive languages which accounts for those languages that have cross-serial dependencies. Since the recognition problem for the class of context-sensitive languages is intractable, we don't want to have to generally use context-sensitive grammars to describe natural languages unless we really have to. What we would ideally like is a grammar that describes only the languages depicted in the set in Figure 2.

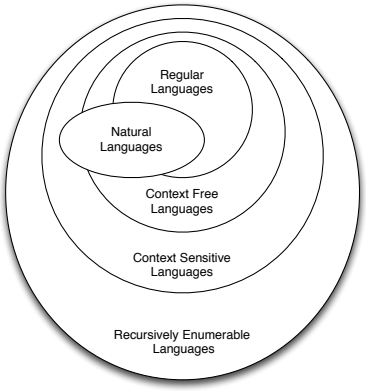


Figure 2: A Venn diagram showing the intersection of the attested natural languages with the Chomsky hierarchy

With this motivation in mind, Joshi [Joshi, 1985] defined a class of languages that is more expressive than context-free languages, less expressive than context-sensitive languages and also sits neatly within the Chomsky hierarchy (thus retaining the properties we already know about). This class of languages is known as the MILDLY CONTEXT-SENSITIVE languages. The abstract language class has the following properties:

- it includes all the context-free languages;
- members of the languages in the class may be recognised in polynomial time;
- the languages in the class account for all the constructions in natural language that context-free languages fail to account for (such as cross-serial dependencies).

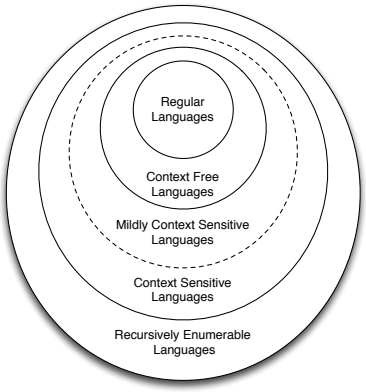


Figure 3: A Venn diagram showing the mildly context sensitive languages within the Chomsky hierarchy

The class of minimally context-sensitive languages is depicted in Figure 3. The grammar that Joshi defined to comply with these properties is called a TREE-ADJOINING GRAMMAR or TAG (see the *Grammars* handout).

For more general information on Formal Language Theory you can try Hopcroft and Ullman [1979] and Rozenberg and Salomaa [1997].

References

N. Chomsky. On certain formal properties of grammars. *Information and control*, 2(2):137–167, 1959.

J.E. Hopcroft and J.D. Ullman, editors. *Introduction to automata theory, languages, and computation*. Reading, MA: Addison-Wesley, 1979.

Aravind K. Joshi. *Tree adjoining grammars: How much context-sensitivity is required to provide reasonable structural descriptions?*, volume http://dx.doi.org/10.1017/CBO9780511597855.007 of *Cambridge Books Online*, pages pp. 206–250. Cambridge University Press, 1985.

G.K. Pullum and G. Gazdar. Natural languages and context-free languages. *Linguistics and Philosophy*, 4(4):471–504, 1982.

G. Rozenberg and A. Salomaa. *Handbook of formal languages: Word, language, grammar*, volume 1. Springer Verlag, 1997.