

Evaluating Subregular Distinctions in the Complexity of Generalized Quantifiers*

Aniello De Santo¹ Thomas Graf² John E. Drury¹
Dept. of Linguistics, Stony Brook University
¹{aniello.desanto, john.drury}@stonybrook.edu
²mail@thomasgraf.net

Abstract

In order to evaluate the complexity of generalized quantifiers, we look at the string languages they can be associated to through the lens of the subregular hierarchy. This approach shows fine-grained complexity distinctions, which we compare to those made by the semantic automata model. The cognitive reality of these predictions will be tested in an eye-tracking study.

1 Introduction

In the study of generalized quantifiers (GQs), it is essential to have an insightful theory of how their meaning is computed. In particular, a sound computational model of quantifier verification should provide a theoretical framework in which to understand the cognitive requirements that have been reportedly associated to different quantified sentences (Lidz et al., 2011; McMillan et al., 2005; Szymanik and Zajenkowski, 2010).

1.1 Semantic Automata

In this perspective, van Benthem’s (van Benthem, 1986) semantic automata (SA) model makes testable predictions about the resources involved in the verification of quantified sentences. Essentially, a generalized quantifier is a function taking two sets A and B , and returning a truth value based on the relation holding between those sets. This relation can be encoded in a binary string, by sending elements of $A \cap B$ to 1 and elements of $A \setminus B$ to 0. Thus, the SA model equates each quantifier Q with a specific string language, and checks whether the language corresponding to $Q(A, B)$ is contained in L_Q . In this perspective, complexity can be measured according the well-known *Chomsky hierarchy of string languages* (Chomsky, 1956):

regular \subsetneq context-free \subsetneq context-sensitive \subsetneq recursively enumerable

Building on this classification, the computational complexity of GQs is then related to the type of automaton that can recognize these languages, and to number of states the automaton needs.

*Presented at the ESSLLI 2017 workshop *QUAD: Quantifiers and Determiners* (Retoré, Steedman)

2 The Subregular Complexity of GQs

Since the automata perspective treats regular (REG) languages as a monolithic unit, the SA model can only explain differences among regular quantifiers by equating complexity to number of states. We argue that a better understanding of these quantifiers can be gained by mapping their string languages to levels in the *Subregular Hierarchy* (SH), which splits REG languages into nested regions of decreasing complexity (McNaughton and Papert, 1971).

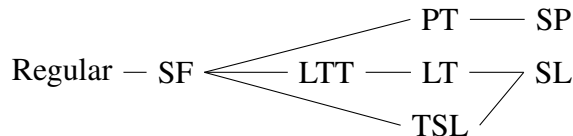
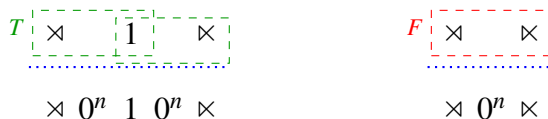


Figure 1: Proper inclusion relationships among subregular language classes.

Consider the sentence *Every A is B*: $every(A,B)$ returns true iff $A \subseteq B$, which implies that the binary string s_{ab} encoding the (A,B) relation contains only 1s. Thus, L_{every} is the string language of just 1s and $every(A,B)$ is true iff $s_{ab} \in L_{every}$. In formal language terms, this can be verified with a *strictly 1-local* (SL) grammar, scanning every symbol in s_{ab} and checking that it is not labelled 0. Another example of SL quantifiers is *No*, associated to the language of all strings with only 0s. At the bottom of the subregular hierarchy, strictly local grammars essentially translate the requirements of a quantifier language into a ban against certain string sequences, encoded as classical n-grams models. However, not all quantifiers can be reduced to this very simple class. Consider now the sentence *Some A are B*, which is true iff there is at least one $b \in A \cap B$. Thus, L_{some} is the language of all strings containing at least one 1. Consider any arbitrary strings $s_1 := 0^n 1 0^n \in L_{some}$ and $s_0 := 0^n \notin L_{some}$. All the n-grams in s_0 are also part of s_1 , so a strictly n -local grammar cannot block 0^n without also incorrectly blocking $0^n 1 0^n$. Thus, L_{some} cannot be SL. But it can be described by a slightly more complex subregular class: the *tier-based strictly local* (TSL) languages. A TSL grammar combines a SL grammar with a mechanism for making specific non-adjacent symbols in the string adjacent via a tier. The main payoff of this property for GQs is the ability to do some limited counting, as checking the occurrence of at least one 1.

Consider again s_0 and s_1 .

Using $\{\times, \times\}$ to mark the start/end of a string, a TSL grammar can express the requirement of L_{some} by disallowing the empty string on the tier of only 1s: $G_{some} = \langle T = \{1\}, S = \{\neg \times \times\} \rangle$. The counting



power of TSL grammars can be extended to any cardinal quantifier. For instance, $L_{at\ most\ 3}$ is obtained by replacing the bigram $\times \times$ in S with the 4-gram 1111 — thus blocking all strings with four or more 1s. Unfortunately, while TSL grammars describe a good number of quantifiers, there are many that cannot be captured this way. The trivial case is that of *proportional quantifiers*, since they are described by languages that are not even regular. Moreover, there are regular GQs that are not TSL. Take *an even number of*. Every string in L_{even} has an even number of 1s, thus a TSL would need to project all 1s, and ban strings with tiers of odd length. This requirement cannot be fulfilled by n -grams constraints. Consider the string $s = 1^n \in L_{even}$. Since $s \in L_{even}$, n must be an even number and the result of adding one more 1 to it will result in the ill-formed string s' . To disallow s' , a TSL grammar G should ban one of the n-grams in s'

from the tier. But s and s' contain exactly the same tier n -grams — $\{\times 1^{n-1}, 1^n, 1^{n-1} \times\}$ — thus G would end up also disallowing s . Since n is arbitrary, no TSL grammar generates L_{even} .

Quantifier	String Constraint	Complexity	Grammar
every	$ s_0 = 0$	SL-1	$\mathbf{T} := \{0, 1\}, \mathbf{S} := \{-0\}$
no	$ s_1 = 0$	SL-1	$\mathbf{T} := \{0, 1\}, \mathbf{S} := \{-1\}$
some	$ s_1 \geq 1$	TSL-2	$\mathbf{T} := \{1\}, \mathbf{S} := \{\neg \times \times\}$
not all	$ s_0 \geq 1$	TSL-2	$\mathbf{T} := \{0\}, \mathbf{S} := \{\neg \times \times\}$
(at least) n	$ s_1 \geq n$	TSL- $(n+1)$	$\mathbf{T} := \{1\}, \mathbf{S} := \{\neg \times 1^k \times\}_{k \leq n}$
(at most) n	$ s_1 \leq n$	TSL- $(n+1)$	$\mathbf{T} := \{1\}, \mathbf{S} := \{\neg 1^{k+1}\}$
all but n	$ s_0 = n$	TSL- $(n+1)$	$\mathbf{T} := \{0\}, \mathbf{S} := \{\neg 0^{n+1}, \neg \times 0^k \times\}_{k \leq n}$
even number	$ s_1 = 2n, n \geq 0$	regular	
most	$ s_1 \geq s_0 $	context-free	

Table 1: Characterization of some GQ in English based on their string complexity.

Note that these characterization results do not depend on the particular order of 0s and 1s in the string. This is crucial, since the way the binary string is built from the encoding of a scene depends on the enumeration adopted by humans during the verification task, and is thus arbitrary.

3 Psycholinguistic Predictions

Clearly, the computational differences highlighted by the subregular approach need to be tested empirically. From a psycholinguistic perspective, this characterization results in specific predictions about the cognitive requirement of different GQs.

$$\underbrace{\{All\}}_{\text{SL}} < \underbrace{\{Some, Less than n, More than n\}}_{\text{TSL}} < \underbrace{\{Even, Odd\}}_{\text{REG}} < \underbrace{\{Less than half, More than half\}}_{\text{CF}}$$

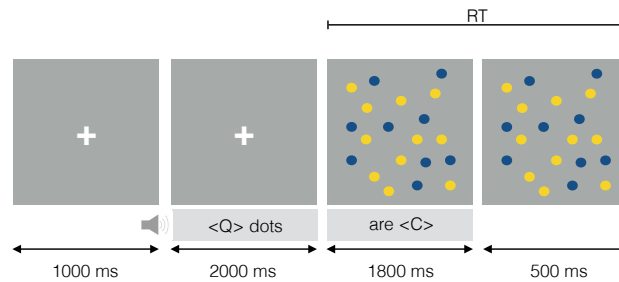
These results partially diverge from what predicted by the SA model:

$$\underbrace{\{All, Some\}}_{\text{FSA}} < \underbrace{\{even, odd\}}_{\text{PSA}} < \underbrace{\{Less than n, More than n\}}_{\text{PSA}} < \underbrace{\{Less than half, More than half\}}_{\text{PSA}}$$

Notably, while the subregular view links the complexity distinctions between parity and cardinal quantifiers to intrinsic properties of the quantifiers themselves, the SA model relies on succinctness claims about the number of states instantiated by the automata generating the quantifier languages. Recent works have extensively shown that — if the use of the approximate number system (ANS) is discouraged by the experimental setup — the ranking predicted by the SA model generally matches working memory predictions (measured as *response times* (RTs)) for verification tasks involving quantified sentences (Szymanik, 2016). However, the link between RTs and cognitive load can be inaccurate, especially when a visual task is involved (Attar et al., 2016). Thus, we plan to evaluate the cognitive complexity of GQs by measuring the increase in subjects’ pupil size during the verification of different quantified sentences. Variations in pupil size have been widely used as an estimate of working memory load in visual search tasks (Just et al., 2003), and have been shown to be sensitive to local resource demands imposed by sentence comprehension (Engelhardt et al., 2010).

3.1 Method

The experimental design and the time course of individual trials are shown in the figure on the right. Participants are asked to judge auditory stimulus sentences of the type $\langle Q \rangle$ of the dots are $\langle Color \rangle$, against a visual display showing systematically varied proportions of two sets of colored dots. The total number of dots in the display is kept constant. The onset of the visual display is delayed until the onset of the disambiguating predicate. This will allow us to measure possible cognitive shifts due just to the encoding of the quantifier.



We made sure that the quantifier onset is equal across all stimuli, and that the onset of the color predicate is the same for each quantifier (Degen and Tanenhaus, 2016). As a comparison to existing results, of which we expect a general replication, RTs are also collected.

4 Conclusions

The subregular classification we presented highlights a finer range of complexity differences among quantifiers. The eye-tracking experiment, almost ready to run, will provide insights into the cognitive reality of these differences. Should the subregular prediction be correct, it would show intriguing parallels to the computational complexity of linguistic dependencies cross-domains (Rogers et al., 2013). A negative result would instead add support to the complexity characterization of GQs as established by the SA model. Given the relative ease of subregular analysis for GQs, such result would also open questions about the actual nature of these computational distinctions.

References

- Attar, N., M. H. Schneps, and M. Pomplun (2016). Working memory load predicts visual search efficiency: Evidence from a novel pupillary response paradigm. *Memory & Cognition* (DOI: 10.3758/s13421-016-0617-8).
- Chomsky, N. (1956). Three models for the description of language. *IRE Transactions on Information Theory* 2, 113–124.
- Degen, J. and M. K. Tanenhaus (2016). Availability of alternatives and the processing of scalar implicatures: A visual world eye-tracking study. *Cognitive Science* 40(1), 172–201.
- Engelhardt, P. E., F. Ferreira, and E. G. Patsenko (2010). Pupillometry reveals processing load during spoken language comprehension. *The Quarterly Journal of Experimental Psychology* 63(4), 639–645.
- Just, M. A., P. A. Carpenter, and A. Miyake (2003). Neuroindices of cognitive workload: Neuroimaging, pupillometric and event-related potential studies of brain work. In *Theoretical Issues in Ergonomics Science*, pp. 56–88.
- Lidz, J., P. Pietroski, J. Halberda, and T. Hunter (2011). Interface transparency and the psychosemantics of most. *Natural Language Semantics* 19(3), 227–256.
- McMillan, C. T., R. Clark, P. Moore, C. Devita, and M. Grossman (2005). Neural basis for generalized quantifier comprehension. *Neuropsychologia* 43(12), 1729–1737.
- McNaughton, R. and S. Papert (1971). *Counter-Free Automata*. Cambridge: MIT Press.
- Rogers, J., J. Heinz, M. Fero, J. Hurst, D. Lambert, and S. Wibel (2013). *Cognitive and Sub-regular Complexity*, Chapter Formal Grammar, pp. 90–108. Springer.
- Szymanik, J. (2016). *Quantifiers and Cognition: Logical and Computational Perspectives*. Springer International Publishing.
- Szymanik, J. and M. Zająkowski (2010). Comprehension of simple quantifiers: Empirical evaluation of a computational model. *Cognitive Science* 34(3), 521–532.
- van Benthem, J. (1986). *Quantifiers*, pp. 25–54. Dordrecht: Springer Netherlands.