Quantifiers satisfying semantic universals are simpler

Iris van de Pol[1] **(i.p.a.vandepol@uva.nl), Paul Lodder**[1] **(paul_lodder@live.nl),**
Leendert van Maanen[2] **(l.vanmaanen@uu.nl), Shane Steinert-Threlkeld**[3] **(shanest@uw.edu),**
Jakub Szymanik[1] **(J.K.Szymanik@uva.nl)**
[1]Institute for Logic, Language and computation, University of Amsterdam,
[2]Experimental Psychology, Utrecht University, [3]Department of Linguistics, University of Washington

Abstract

Despite wide variation among natural languages, there are linguistic properties thought to be universal to all or almost all natural languages. Here, we consider universals at the semantic level, in the domain of quantifiers, which are given by the properties of *monotonicity*, *quantity*, and *conservativity*. We investigate whether these universals might be explained by differences in complexity. We generate a large collection of quantifiers, based on a simple yet expressive grammar, and compute both their complexities and whether they adhere to these universal properties. We find that quantifiers satisfying semantic universals are less complex: they have a shorter minimal description length.

Keywords: semantic universals; generalized quantifiers; logical grammar; complexity; minimal description length

Introduction

If you have ever tried to learn a new language, you will know that this can be a challenge. You have to learn a lot of new things that are different from the language you are used to. While the world's languages have many differences, at the same time, interestingly, most languages also share a striking amount of similarities, called linguistic universals (Croft, 1990; Goddard & Wierzbicka, 2002; Greenberg, 1966). Here we study such universals at the semantic level, in the domain of quantifiers (Barwise & Cooper, 1981; von Fintel & Matthewson, 2008).

Quantifiers are semantic objects that express quantitative relational properties, such as `some`, `most`, or `all`. It has been observed that the quantifiers that are lexicalized (as monomorphemic words) in natural language share certain structural properties, namely those of *monotonicity*, *quantity*, and *conservativity* (Barwise & Cooper, 1981; Keenan & Stavi, 1986). When looking at the space of all logically possible quantifiers, however, a large majority does not have these properties.

The question arises why these universals hold. Why do quantifiers in natural language have precisely these properties? A possible explanation for these universals lies in the interaction between these properties and our cognitive apparatus (see, e.g., Gibson et al., 2019, Steinert-Threlkeld & Szymanik, 2020). This builds on the idea that the human cognitive system is structured in such a way that it favors dealing with certain meanings over others.

In this paper, we explore the following hypothesis: quantifiers having these semantic properties are *simpler*. Simplicity as an explanatory concept in cognition has been studied in a variety of domains (Chater & Vitányi, 2003; Feldman, 2016). Here we measure simplicity by the concept of minimal description length in a logical grammar (i.e., a language of thought), a framework that has been used, e.g., in the domain of concept learning (Feldman, 2000; Goodman, Tenenbaum, Feldman, & Griffiths, 2008), language acquisition (Piantadosi, Tenenbaum, & Goodman, 2013), and auditory memory (Planton et al., 2021). We use this grammar to generate a collection of over 24,000 logically possible quantifiers, and we compute both their complexities and their adherence to universal properties. Using a logistic regression model, we show that quantifiers with universal properties are simpler.

This work extends the small-scale study by Van de Pol, Steinert-Threlkeld, and Szymanik (2019), that used a measure from the framework of algorithmic information theory—in particular, an approximation to Kolmogorov complexity (Li & Vitányi, 2008) based on the Lempel-Ziv compression algorithm (Lempel & Ziv, 1976)—and tested this for a handful of quantifier pairs. Their study uses a minimal-pair methodology to select pairs of quantifiers that are similar but differ with respect to their adherence to universal properties—a method that is, unfortunately, not scalable. We overcome these limits of scale by generating a large collection of quantifiers in a principled manner and analyzing the association between complexity and universal property for the entire space of quantifiers. Furthermore, to test the robustness of their methods we also compute and analyse the Lempel-Ziv complexity of all quantifiers in our study and compare results.

The paper is structured as follows. First, we introduce the framework of generalized quantifiers and the properties of monotonicity, quantity, and conservativity, and the semantic universals in relation to these properties. Then, we define the logical grammar that we use to generate a formal language of quantifiers, and we describe the methods that we use to compute their minimal description lengths and their Lempel-Ziv complexities. We report on both descriptive statistics and bootstrapped logistic regression results, by which we analyse the relation between quantifiers satisfying universal properties and those that do not, and their level of simplicity. Finally, we compare our results to previous complexity results by Van de Pol et al. (2019) and previous learnability results by Steinert-Threlkeld and Szymanik (2020) and we make suggestions for future research.

Quantifiers

Quantifiers are the semantic objects that are expressed by (quantified) determiners,[1] such as `some`, `most`, or `all but one`, which describe quantity in a noun phrase. Determiners are expressions that can combine with common nouns and a verb phrase in simple sentences of the form *Det N VP*, like "some bicycles are red." We assume a division between simple and complex determiners, where `some` and `most` are examples of simple determiners and `all but one` is an example of a complex determiner. Roughly speaking, one can think about this as a division between quantifiers that are lexicalized (as mono-morphemic words) and those that are not (Barwise & Cooper, 1981; Keenan & Stavi, 1986).

Generalized Quantifiers

We use the framework of generalized quantifiers to represent the meaning of quantifiers as a collection of models.[2] A model consists of a collection of objects and their properties. It can be seen as a snapshot of a particular part of the world. For instance, for the expression "some bicycles are red," we look at the collection of bicycles and the collection of red things, and when those overlap we know that there are some bicycles that are red.

Formally, a model $\mathcal{M} = \langle M, A, B \rangle$ consists of a domain, the set M, and two subset of that domain, sets $A, B \subseteq M$, where the sets A and B possibly overlap. Functionally, M is called the domain of discourse, A is called the restrictor, and B is called the scope. The sentence "some bicycles are red" then means that there are some objects in A that are also in B. In addition, we enumerate the objects in the domain, giving us an ordering $<$ over M, which allows to model sentences in which the order of the objects matter, such as "the first 3 bicycles are red." See Figure 1 for an illustration of a model of the form $\mathcal{M} = \langle M, A, B, < \rangle$.

A quantifier can then be represented and defined by a quantifier expression: a formula in a set-theoretic language. The quantifier is the collection of all models in which that quantifier expression is true. For instance, the meaning of the quantifier `all` can be represented by the expression $A \subseteq B$. Formally, we would express the meaning of `all` by $[\![\texttt{all}]\!] = \{\langle M, A, B \rangle : A \subseteq B\}$.

Universal Properties of Quantifiers

Using the framework of generalized quantifiers we can define various properties. A quantifier is *monotone*[3] when it is either *upward* or *downward monotone*. Monotonicity expresses that
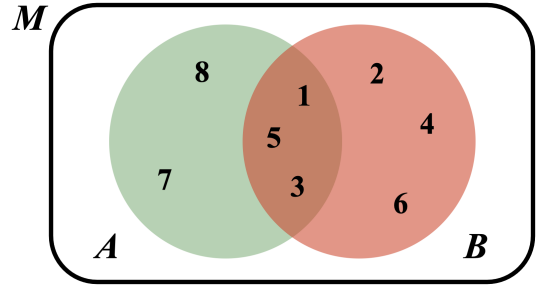

Figure 1: An example of a model of the form $\mathcal{M} = \langle M, A, B, < \rangle$. The numbers represent the objects in the domain and they correspond to the position of the objects in the order $<$. The sentence "the first 3 bicycles are red" is true in this model, as the first three objects in A (the objects with position 1, 3, and 5 in the order $<$ over the whole domain) are also in B.

the meaning of a quantifier does not change when expanding or contracting its scope, i.e., the set B. Formally, it is defined as follows. Let Q be a generalized quantifier over ordered models.[4] Q is *upward monotone* when: if $\langle M, A, B, < \rangle \in$ Q and $B \subseteq B'$, then $\langle M, A, B', < \rangle \in$ Q. Analogously, Q is *downward monotone* when: if $\langle M, A, B, < \rangle \in$ Q and $B' \subseteq B$, then $\langle M, A, B', < \rangle \in$ Q. For example, the quantifier `some` is upward monotone, `few` is downward monotone, and `exactly two` is neither upward nor downward monotone and therefore non-monotone.[5] The following universal has been proposed with respect to monotonicity:

- *All simple (quantified) determiners express monotone quantifiers* (Barwise & Cooper, 1981).[6]

The property of *quantity*[7] expresses that the meaning of a quantifier only depends on the sizes of $A \cap B, A \setminus B, B \setminus A$, and $M \setminus (A \cup B)$. In contrast, its meaning does not depend on the ordering of the objects in the domain or on their individual identities or names. Formally, we say that Q is *quantitative* when: if $\langle M, A, B, < \rangle \in$ Q and for M', A', B' with $A', B' \subseteq M'$ it holds that $A \cap B, A \setminus B, B \setminus A$, and $M \setminus (A \cup B)$ have the same cardinalities as $A' \cap B', A' \setminus B', B' \setminus A'$, and $M' \setminus (A' \cup B')$, and $<'$ is an order over M', then $\langle M', A', B', <' \rangle \in$ Q. For example, the quantifier `three` is quantitative, and the quantifier `the first 3` is not quantitative. The following universal has been proposed with respect to quantity:

---

[1]Not all determiners express quantifiers. For instance, the demonstrative determiners—like *those*—do not express a quantifier. We refer to determiners that express quantifiers as quantified determiners.

[2]Specifically, we use the framework of type $\langle 1, 1 \rangle$ generalized quantifiers. For a textbook treatment of generalized quantifiers see Peters and Westerståhl (2006). For more details on computational representations of quantifiers, see Szymanik (2016)

[3]We use the general term *monotone* to refer to what is also called *right monotone* for type $\langle 1, 1 \rangle$ quantifiers and *monotone in the second argument* for arbitrary quantifiers. In our case (that of type $\langle 1, 1 \rangle$ quantifiers) this means that a quantifier is monotone in the set B.

[4]Note that the order $<$ over the models plays no role in the definition of monotonicity. Neither does it play a role in the definition in conservativity. Since it does play a role in the definition of the quantity property, we include it for the sake of uniformity of presentation.

[5]We consider the literal meaning of quantifiers, not including their implicatures or presuppositions.

[6]In fact, the original claim by Barwise and Cooper (1981) is a bit weaker, including not just monotone quantifiers, but also conjunctions of monotone quantifiers.

[7]The term *quantity* was introduced by Van Benthem (1984), it refers to the same property that is called *logical* by Keenan and Stavi (1986) and *isomorphism closure* by Peters and Westerståhl (2006).

- *All simple (quantified) determiners express quantitative quantifiers* (Keenan & Stavi, 1986).

The property of *conservativity*[8] expresses that to verify a quantifier, the objects in B that are not in A are not relevant: the meaning of such a quantifier is really about the noun and less so about the verb phrase. Formally, we say that Q is *conservative* when: $\langle M, A, B, < \rangle \in$ Q if and only if $\langle M, A, A \cap B, < \rangle \in$ Q. For example, the quantifier most is conservative, and exactly as many A's as B's is not conservative. The following universal has been proposed with respect to conservativity:

- *All simple (quantified) determiners express conservative quantifiers* (Barwise & Cooper, 1981; Higginbotham & May, 1981; Keenan, 1981; Keenan & Stavi, 1986).[9]

We interpret these universals as constraints on language in the form of general tendencies or biases, not as fully strict demarcations. An extensive discussion and defense of these universals falls outside of the scope of the current study. Instead, we focus on how these properties relate to the complexity or simplicity of quantifiers and whether simplicity could explain the pervasiveness of these properties in natural language.

Methods

We use the following methods to generate a large body of generalized quantifiers, in order to study their complexities in relation to the universal properties being present or not.

Grammar and Language

To study the universal properties of quantifiers we need to look both at quantifiers that do and that do not have these properties. We use a principled way of generating a large space of generalized quantifiers with and without these properties. In particular, we use a logical grammar to generate quantifier expressions, over which the meaning of a quantifier[10] can be computed.

We define a simple yet expressive grammar that consists of basic building blocks and standard rules for how to combine them. In particular, the grammar is defined by the collection of operators presented in Table 1. These consist of standard set-theoretic operators $(\cup, \cap, \backslash, | \cdot |, \subseteq)$, integer operators $(=, >)$, and boolean operators $(\neg, \wedge, \vee)$. In order to investigate the property of *quantity*, we need an operator that is sensitive to the ordering over objects. In fact, whenever we say set, we mean a tuple of a set and an ordering $<$ over the elements in the set. We include such an index-like operator, namely $\iota(\cdot, \cdot)$,

operator	type	gloss
$\cup$	SET $\times$ SET $\to$ SET	union
$\cap$	SET $\times$ SET $\to$ SET	intersection
$\backslash$	SET $\times$ SET $\to$ SET	setminus
$\iota(\cdot, \cdot)$	INT $\times$ SET $\to$ SINGLETON SET	'object at index'
$\| \cdot \|$	SET $\to$ INT	cardinality
$\subseteq$	SET $\times$ SET $\to$ BOOL	subset equal
$=$	INT $\times$ INT $\to$ BOOL	integer equality
$>$	INT $\times$ INT $\to$ BOOL	integer larger than
$\neg$	BOOL $\to$ BOOL	negation
$\wedge$	BOOL $\times$ BOOL $\to$ BOOL	and
$\vee$	BOOL $\times$ BOOL $\to$ BOOL	or

Table 1: The collection of operators used to generate quantifier expressions. Note that the sets are ordered, which is of relevance for operator ι.

that, given a set and an index position, returns a singleton with the object at that index position, given the ordering over the set.

In addition to the operators, the grammar has two variables $\{A, B\}$—where A and B are placeholders for sets (with an ordering)—and a collection of constants $\{0, 1, \ldots, s\}$,[11] which represent integers. This grammar defines a formal language with quantifier expressions, which are the expressions that return a Boolean value and that can be formed by using the given variables, constants, and operators, adhering to the type restrictions of the operators.

Minimal Expression Length

Quantifier expressions in this language are not unique. The quantifier at most one can, for instance, be defined both by $(2 > |A \cap B|)$ and by $\neg (|A \cap B| > 1)$. The meanings of these expressions are equivalent: they are true in exactly the same models. We define the length of an expression by the number of operators in it. So the length of expression $(2 > |A \cap B|)$ is three and the length of expression $\neg (|A \cap B| > 1)$ is four. The minimal expression length of a quantifier in this language is the length of the shortest expression for this quantifier.

We generate the collection of quantifier expressions by the following procedure. We first generate all expressions of length 1, one by one, by going through the list of operators. For each expression we compute its meaning for all models from size 1 to size s. We compare this meaning to the meanings of expressions that we stored so far. If the meaning is not yet present, we add this expression and its meaning to our collection. If the meaning was already present, this means we already included an equivalent expression of equal or shorter length. Then we do not add it and continue with the next quantifier expression in line. When finished with all possible quantifiers of length one, we continue with quantifiers of

---

[8]The term *conservativity* was introduced by Keenan (1981), it refers to the same property that is called *lives on* by Barwise and Cooper (1981), and *intersectivity* by Higginbotham and May (1981).

[9]In fact, the original claim is even stronger, namely that *all* (quantified) determiners express conservative quantifiers, not just the simple ones. See also Zuber and Keenan (2019) for an alternative definition of the conservativity constraint.

[10]The meaning of a quantifier is defined by the collection of models in which a quantifier expression is true. This is also called the *extension* of a quantifier.

[11]Where s is the maximum model size that is considered, which, to limit computational blow-up, we set to 8.

length two and repeat the procedure up to length ℓ. This way, we generated all 24,632 semantically unique quantifiers expressions, defined by this grammar, up until expression length five.[12] By virtue of this procedure, we know that all quantifier expressions in this language are of minimal expression length: their meanings cannot be expressed by a shorter combination of operators in our grammar.

Encoding Quantifier Meanings

To compare the meanings of the quantifier expressions, we generate binary representations of those meanings. This works as follows. We first encode each model as a sequence of symbols. We give the different subareas in a model a label, say $A \cap B \mapsto d, A \setminus B \mapsto e$, and $B \setminus A \mapsto f$,[13] we label the objects by their area, and place each label in a sequence, based on the order of the objects. The order of those labels in the sequence represents the order over the objects. For example, the model in Figure 1 is encoded by $dfdfdfee$. Then, we enumerate all models from small to large, up to a maximum model size s, in a fixed order, in particular, in the lexicographic order over the encodings of the model, i.e., the dictionary order over the labels (d, e, f).[14] Finally, for each of the models in the sequence, we put a 1 when the quantifier expression is true in that model and a 0 otherwise. This results in a unique representation for each quantifier meaning, up to a certain model size. Unsurprisingly, computationally we can only deal with finite sequences, and given the exponential blow-up of the binary encodings of quantifiers in terms of the maximum model size s, we limit the maximum model size to eight.

In addition to minimal expression length, we also compute the Lempel-Ziv complexity of each quantifier, for comparison with the results of the study by Van de Pol et al. (2019). This complexity measure is computed over the binary encoding of the quantifier meanings, using the Lempel-Ziv compression algorithm, which measures the number of unique subpatterns, when scanning the string from left to right. For further details, we refer to Van de Pol et al. (2019).

Results

Using the described procedures we generated a collection of 24,632 quantifiers. For each quantifier we computed whether they have the property of monotonicity, quantity, and conservativity, and we computed their complexity scores, both for minimal expression length (ML) and Lempel-Ziv complexity (LZ). To facilitate the comparison of the results for ML and LZ, we normalized the complexity data by computing their z-scores. The code that we used for generating these data and

---

[12]The number of quantifier expressions grows exponentially in the maximum expression length ℓ. To manage this exponential blow-up we limited the maximum expression length to five.

[13]We are assuming the property of *extensionality*, i.e., that the subarea $M \setminus (A \cup B)$ does not matter for the meaning of a quantifier (Szymanik, 2016). Therefore the four subareas in Figure 1 are now reduced to three.

[14]In principle, any fixed ordering of the models can be used for this.

	YES	NO
monotonicity	-0.12	0.05
quantity	-0.15	0.03
conservativity	-0.16	0.02

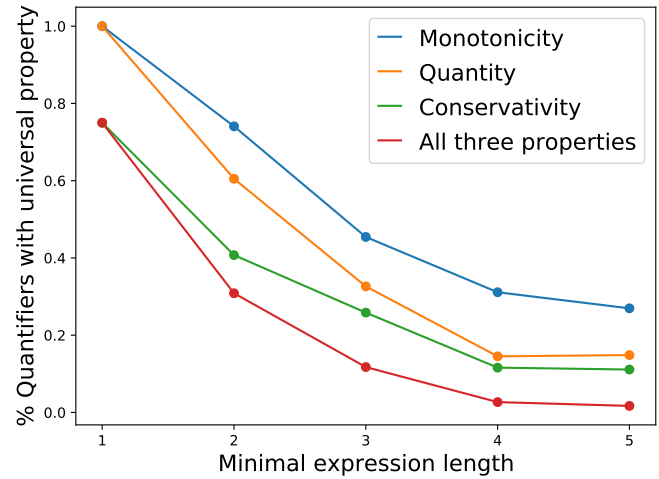Table 2: Averge normalized ML scores of quantifiers with versus without universal property.

Figure 2: Percentage of universal property per minimal expression length.

the data themselves can be found at `https://github.com/ivdpol/QuantifierComplexity/tree/CogSci2021`.

Descriptive Statistics

The descriptive statistics show a negative relation between minimal expression length (ML) and universal property. For each universal property, the average ML of quantifiers with that property is consistently lower than the average ML of quantifiers without that property. See Table 2. The fact that these difference are only small is unsurprising given the high total average of raw ML scores (of 4.84) over all quantifiers (ML scores range between 1 and 5).[15] Furthermore, for each universal property, the average level of universal property shows a negative relation to ML: the lower the minimal expression length, the higher the proportion of quantifiers with a universal property. See Figure 2.

Logistic Regression Results

In addition to considering the descriptive statistics, we performed a logistic regression for each universal property, with the universal property as dependent variable and complexity as the independent variable. We used bootstrap resampling to compute a distribution over the regression coefficient, and we compared this to a baseline, which we generated by ran-

---

[15]This high total average follows from the fact that the number of different expressions grows exponential in expression length.

domly shuffling the actual complexity values over the different quantifiers, each time before taking a sample. We computed the coefficient for the original complexity data and the randomly shuffled complexity data in pairs, over the same random samples of quantifiers.[16] The regression results for all three universal properties that we looked at show a negative relation between ML complexity and the universal property, i.e., a positive relation between simplicity and universal property. See Figure 3.

Monotonicity The logistic regression results for monotonicity show a negative relation between ML and monotonicity. The difference (per sample) between the coefficient of the original (normalized) ML data and the coefficient of the randomly shuffled (normalized) ML scores has a mean of -0.15 (95% CI [-0.24, -0.06]). See Figure 3.

Quantity The logistic regression results for quantity show a negative relation between ML and quantity. The difference (per sample) between the coefficient of the original (normalized) ML data and the coefficient of the randomly shuffled (normalized) ML scores has a mean of -0.15 (95% CI [-0.26, -0.04]). See Figure 3.

Conservativity The logistic regression results for conservativity show a negative relation between ML and conservativity. The difference (per sample) between the coefficient of the original (normalized) ML data and the coefficient of the randomly shuffled (normalized) ML scores has a mean of -0.15 (95% CI [-0.28, -0.03]). See Figure 3.

Summary For all three semantic universals, these regressions show that, in general, quantifiers satisfying the universal are simpler than those that do not: they have a shorter minimal description length.

Lempel-Ziv Complexity

In addition to minimal description length, we also measured Lempel-Ziv complexity, as used in (Van de Pol et al., 2019), and ran the same regressions as for minimal description length. The results for Lempel-Ziv complexity show a negative relation between LZ and monotonicity, a positive relation between LZ and conservativity, and no relationship between LZ and quantity, both for the descriptive statistics and the bootstrapped logistic regression results. The average (normalized) LZ complexity for quantifiers with versus without the universal property is -0.19 versus 0.08 for monotonicity, 0.18 versus -0.02 for conservativity, and 0.03 versus 0.00 for quantity. The logistic regression results are as follows. For monotonicity, the difference (per sample) between the coefficient of the original (normalized) LZ data and the coefficient of the randomly shuffled (normalized) LZ scores has a mean of -0.28 (95% CI

---

[16]We use a sample size of 5,000 quantifiers and we repeat the proces for 20,000 random samples.

[-0.37, -0.18]). For conservativity, the difference (per sample) between the coefficient of the original (normalized) LZ data and the coefficient of the randomly shuffled (normalized) LZ scores has a mean of 0.20 (95% CI [0.08, 0.33]). For quantity, the difference (per sample) between the coefficient of the original (normalized) LZ data and the coefficient of the randomly shuffled (normalized) LZ scores has a mean of 0.03 (95% CI [-0.09, 0.15]).

Discussion

We investigated the complexity of quantifiers in relation to semantic universals. We studied whether a bias towards simplicity could explain the semantic universals of monotonicity, quantity, and conservativity. We used the framework of generalized quantifiers and a simple yet expressive logical grammar (language of thought) to generate a formal language of over 24,000 quantifier expressions. For each of these quantifiers we computed its minimal expression length and whether it has the properties of monotonicity, quantity, and conservativity. We found for each of these universal properties that quantifier expressions that satisfy them are simpler: they have a shorter minimal expression length. This suggests the following explanation for semantic universals in the domain of quantifiers: meanings satisfying semantic universals are simpler.

We used the framework of generalized quantifiers because it is a well-defined and well-studied framework for representing the meaning of quantifiers. Our aim in defining the grammar that we used to build a large collection of quantifier expressions, was to keep it as basic as possible, while at the same time capturing a significant part of natural language quantifiers and in addition also going beyond natural language. Where possible, we avoided complex operators that are combinations of more basic operators, thereby not including single operators for, i.e., "is an empty set" or "is of an even number." Since there are multiple collections of basic set-theoretical and logical operators that are definable in terms of each other, i.e., that in the infinite case define the same collection of expressions, there is not just one unique grammar that satisfies these objectives. Future work includes investigating such alternative grammars and comparing the results.

The descriptive statistics showed that there are quite a few expressions with the relatively short expression length of two, that do not satisfy one or more universal properties. The majority of these expressions include the ι operator—which takes as input and integer i and a set P and returns a singleton with the the i-th object in P—and all but one of these expressions include an integer constant. For example, for $i \in \{1,\ldots,8\}$ the expression $A \subseteq \iota(i, B)$ is an expression in our language that has length two and that does not satisfy monotonicity, quantity, or conservativity. To our knowledge, there is no quantifier attested in natural language that expresses this meaning, which could be described by "either there is no A or there is exactly one A, which is the i-th B." The prevalence of the ι operator in these expressions suggests that the the ι operator might be a less basic operator. Future work could include refining the
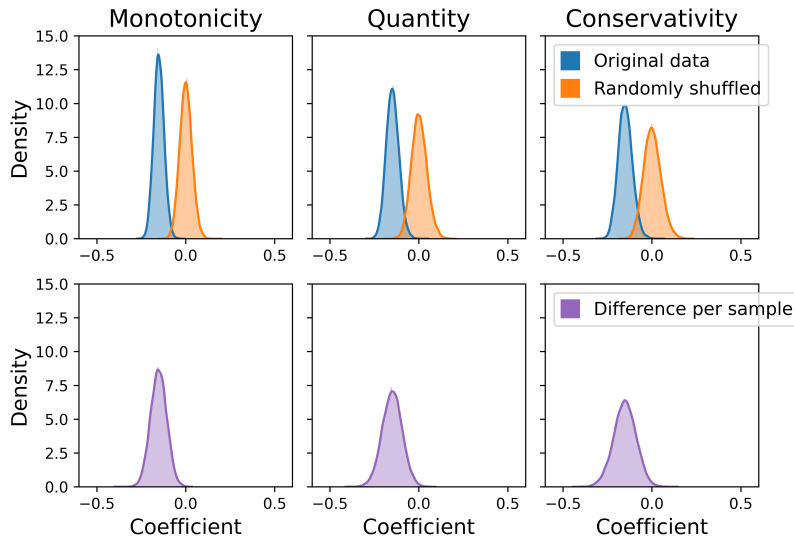
Figure 3: Bootstrapped logistic regression results for normalized ML scores.

definition of expression length by assigning different weights to the operators, and possibly giving extra weight to the ι operator.

Both the descriptive and logistic regression results for minimal expression length show a robust but relatively small difference in complexity between quantifiers with versus without the universal properties. This suggests on the one hand that a bias for simplicity might indeed be an explanatory factor for these semantic universals and, on the other hand, that simplicity is likely not the only force at play in shaping the semantic properties of quantifiers. Other likely candidates that could play a role in either pushing towards or away from these properties are cultural evolution (Carcassi, Steinert-Threlkeld, & Szymanik, 2019) and communicative needs (Steinert-Threlkeld, 2020).

In addition to the minimal expression length, we also computed the Lempel-Ziv complexities of the quantifiers, for comparison with the results of the study by Van de Pol et al. (2019). For monotonicity, the LZ results for our collection of quantifiers were in line both with the ML results and with the LZ results in the study by Van de Pol et al. (2019)—that looked at a small collection of quantifiers, and compared between minimally differing pairs of quantifiers satisfying and not satisfying a universal. Monotone quantifiers were found to have a lower complexity in all of these three cases. For conservativity, however, the LZ results were neither in line with the ML results that we found in the current study, nor with the LZ results in the study by Van de Pol et al. (2019). Their study showed no effect for the relation between LZ and conservativity, while our study found opposite effects for LZ and ML with respect to conservativity: conservative quantifiers have a lower ML but a higher LZ than non-conservative quantifiers. Also for quantity, the LZ results were neither in line with the current ML results nor with the LZ results in the study by Van de Pol et al. (2019). While their results for LZ in relation to quantity

were not fully robust, they did find a tendency towards quantitative quantifiers being less complex. Our LZ results, however, showed no effect for the relation to conservativity, while our ML results, on the other hand, do show the effect that quantitative quantifiers are less complex, i.e., have a lower ML score. Given the differences between the LZ results for the current large-scale study and the LZ results for the small-scale study by Van de Pol et al. (2019), we conclude that the Lempel-Ziv complexity as a measure of complexity for quantifiers in the context of semantic universals does not scale up robustly.

It is commonly expected that simplicity and learnability will correlate (Chater & Vitányi, 2003), and it is plausible that when a quantifier is simpler it is easier to learn. For monotonicity and quantity these simplicity results indeed agree with the learnability results in the study by Steinert-Threlkeld and Szymanik (2020). They found that monotone quantifiers are easier to learn by a recurrent neural network than non-monotone quantifiers, and idem for quantitative quantifiers. For conservativity, they found no difference in learnability for conservative versus non-conservative quantifiers. It is an ongoing debate whether conservative quantifiers are indeed easier to learn. While Hunter and Lidz (2013) found that conservative quantifiers were easier to learn for children, this effect was not found in a recent replication of their study by Spenader and de Villiers (2019).

To further corroborate these results, future work would ideally scale up our methods even further, both in terms of the maximum expression length of the quantifiers that are being considered, as well as in terms of the maximum model size over which the meanings of the quantifiers are evaluated. Given the computational blow-up inherent in the methods used, it is not trivial how such scaling could be achieved. For further comparison between the simplicity and learnability of quantifiers in the context of semantic universals, similar methods as used by Steinert-Threlkeld and Szymanik (2020)

could be used to compute the learnability of the collection of quantifiers that we considered here. Finally, one could extend the present framework to other domains than quantifiers and possibly induce a maximally explanatory grammar over multiple domains.

Acknowledgements

References

Barwise, J., & Cooper, R. (1981). Generalized Quantifiers and Natural Language. *Linguistics and Philosophy*, *4*(2), 159–219.

van Benthem, J. (1984). Questions about quantifiers. *The Journal of Symbolic Logic*, *49*(2), 443–466.

Carcassi, F., Steinert-Threlkeld, S., & Szymanik, J. (2019). The emergence of monotone quantifiers via iterated learning. In *Proceedings of cogSci 2019*.

Chater, N., & Vitányi, P. (2003). Simplicity: A unifying principle in cognitive science? *Trends in Cognitive Sciences*, *7*(1), 19–22.

Croft, W. (1990). *Typology and Universals*. Cambridge: Cambridge University Press.

Feldman, J. (2000). Minimization of boolean complexity in human concept learning. *Nature*, *407*(6804), 630–633.

Feldman, J. (2016). The simplicity principle in perception and cognition. *Wiley Interdisciplinary Reviews: Cognitive Science*, *7*(5), 330–340.

Gibson, E., Futrell, R., Piantadosi, S. T., Dautriche, I., Mahowald, K., Bergen, L., & Levy, R. (2019). How Efficiency Shapes Human Language. *Trends in Cognitive Sciences*.

Goddard, C., & Wierzbicka, A. (2002). *Meaning and universal grammar: Theory and empirical findings* (Vol. 1). John Benjamins Publishing.

Goodman, N. D., Tenenbaum, J. B., Feldman, J., & Griffiths, T. L. (2008). A rational analysis of rule-based concept learning. *Cognitive science*, *32*(1), 108–154.

Greenberg, J. H. (1966). *Language Universals*. The Hague: Mouton.

Higginbotham, J., & May, R. (1981, 1). Questions, quantifiers and crossing. *The Linguistic Review*, *1*, 41-80.

Hunter, T., & Lidz, J. (2013). Conservativity and learnability of determiners. *Journal of Semantics*, *30*(3), 315–334.

Keenan, E. L. (1981). A boolean approach to semantics. In J. A. Groenendijk, T. M. Janssen, & M. B. Stokhof (Eds.), *Formal methods in the study of language: Part 2* (pp. 343–379). Amsterdam: Mathematisch Centrum.

Keenan, E. L., & Stavi, J. (1986). A semantic characterization of natural language determiners. *Linguistics and Philosophy*, *9*(3), 253–326.

Lempel, A., & Ziv, J. (1976). On the complexity of finite sequences. *IEEE Transactions on Information Theory*, *22*(1), 75–81.

Li, M., & Vitányi, P. (2008). *An Introduction to Kolmogorov Complexity and Its Applications*. Springer.

Peters, S., & Westerståhl, D. (2006). *Quantifiers in Language and Logic*. Oxford: Clarendon Press.

Piantadosi, S. T., Tenenbaum, J. B., & Goodman, N. D. (2013). *Modeling the acquisition of quantifier semantics: A case study in function word learnability*.

Planton, S., van Kerkoerle, T., Abbih, L., Maheu, M., Meyniel, F., Sigman, M., . . . Dehaene, S. (2021). A theory of memory for binary sequences: Evidence for a mental compression algorithm in humans. *PLOS Computational Biology*, *17*(1), e1008598.

van de Pol, I., Steinert-Threlkeld, S., & Szymanik, J. (2019). Complexity and learnability in the explanation of semantic universals of quantifiers. In *Proceedings of Cogsci 2019*.

Spenader, J., & de Villiers, J. (2019). Are conservative quantifiers easier to learn? evidence from novel quantifier experiments. In J. J. Schlöder, D. McHugh, & F. Roelofsen (Eds.), *Proceedings of the 22nd Amsterdam Colloquium*.

Steinert-Threlkeld, S. (2020). Quantifiers in natural language optimize the simplicity/informativeness trade-off. In J. J. Schlöder, D. McHugh, & F. Roelofsen (Eds.), *Proceedings of the 22nd Amsterdam Colloquium* (pp. 513–522).

Steinert-Threlkeld, S., & Szymanik, J. (2020). Ease of learning explains semantic universals. *Cognition*, *195*, 104076.

Szymanik, J. (2016). *Quantifiers and cognition. logical and computational perspectives*. Springer.

von Fintel, K., & Matthewson, L. (2008). Universals in semantics. *The Linguistic Review*, *25*(1-2), 139–201.

Zuber, R., & Keenan, E. L. (2019). A note on conservativity. *Journal of Semantics*, *36*(4), 573–582.