

Complex simplex numerals

Marcin Wągiel & Pavel Caha

Abstract

It is commonly assumed that basic cardinal numerals such as English *three* are simplex expressions whose primary function is to quantify over entities denoted by the modified NP (e.g., Kennedy 2015; Rothstein 2017; Ionin and Matushansky 2018). In this paper, we explore cross-linguistic marking patterns suggesting that cardinals in fact lexicalize complex syntactic and semantic structures derived from the primitive notion of the number scale. The evidence we will investigate comes from various morphological shapes of cardinal numerals when used to count objects and when used for abstract arithmetical counting.

Keywords: cardinal numerals, typology, morphosemantics, Nanosyntax

1 Introduction

Though cardinals can be used in various ways (for an overview, see e.g., Bultinck 2005), their most widely studied function is to enumerate entities designated by the noun. For instance, the numeral *three* in sentences such as (1) is used as a prenominal modifier which quantifies over individuals in the denotation of the modified NP. In (1-a) it specifies the number of apples that fell from the table, whereas in (1-b) the total number of the relevant musketeers. We will refer to this function as OBJECT COUNTING.

- (1) a. **Three** apples fell from the table.
- b. The **three** musketeers fought bravely.

The second function of cardinal numerals we will investigate here concerns reference to an abstract number concept (Bultinck 2005; Rothstein 2017). In sentences such as those in (2), the numeral *three* does not enumerate any objects but rather it refers to an abstract mathematical entity which can enter arithmetical relations, see (2-a), and have properties such as parity, as in (2-b). We will call this second function ABSTRACT COUNTING.

- (2) a. Five minus **three** is two.
- b. **Three** is an odd number.

Conceptually, the distinction between object and abstract counting is evident. However, given that the very same morphological form is used in both (1) and (2), it is not immediately clear whether the distinction is also relevant for the grammar of natural languages. In this paper,

we argue that it is indeed relevant for the grammar and we put forth a specific relationship between object-counting and abstract-counting uses of numerals.

In order to determine the underlying meaning components (features) of numerals and the morphosyntactic structures they give rise to, we explore six types of meaning/form correspondences attested across languages that pertain to the distinction between object counting and abstract counting. Based on this cross-linguistic evidence, we argue that all (apparently simplex) cardinals are in fact morphosyntactically complex expressions. We shall further propose that object-counting numerals are both syntactically and semantically derived from the abstract-counting numerals.

The paper is structured as follows. In Section 2, we discuss evidence for the grammatical relevance of abstract and object counting in natural language along with some existing approaches to this distinction. In Section 3, we examine morphologically simplex abstract-counting numerals. We show that such numerals exhibit three distinct types of morphological relations to their object-counting counterparts (syncretism, containment, suppletion). Section 4 provides evidence that the same patterns of relationship also apply to numerals that are morphologically complex in their abstract-counting function. In Section 6, we propose a compositional analysis of the meaning of abstract- and object-counting numerals which builds on the idea of invariant semantic primitives. In Section 7, we demonstrate how the proposed system can derive various marking patterns discussed in Section 3 and 4. Section 8 concludes.

2 Two functions of cardinals

From the fact that there is a conceptual distinction between abstract counting and object counting, it does not necessarily follow that this distinction is reflected in the grammar of natural languages. However, there is evidence that distinguishing between abstract- and object-counting uses of cardinals is linguistically relevant. The crucial contrasts amount to the fact that object-counting numerals denote counting devices, i.e., operations allowing for numeric quantification over objects, whereas abstract-counting numerals simply refer to arithmetical entities.

2.1 Linguistic relevance

It has been observed in the literature that abstract number concepts can have different properties than pluralities of individuals (Rothstein 2017). Some examples include being prime, odd or a natural number, see (3). On the other hand, such properties cannot be predicated of pluralities. For instance, (4-a) and (4-b) have very different truth conditions than the corresponding assertions in (3-a) and (3-b), whereas (4-c) is simply infelicitous.

- (3) a. **Three** is prime.
b. **Three** is odd.
c. **Three** is a natural number.
- (4) a. **#Three** things are prime.
b. **#Three** things are odd.

- c. #**Three** things are a natural number.

Furthermore, mathematical statements such as those in (5) require abstract-counting numerals. As evidenced by the infelicity of (6), even numeral phrases including the virtually vacuous noun *thing* are incompatible with mathematical environments which call for numeric values and not pluralities of objects.

- (5) a. **Three** times two equals six.
 b. Six divided by **three** equals two.
- (6) a. #**Three** things times two things equals six things.
 b. #Six things divided by three things equals two things.

In addition, certain grammatical constructions require arguments referring to number concepts rather than to pluralities. For instance, only abstract-counting numerals can be complements of *count (up) to*, as witnessed by the contrast between (7-a) and (8-a). Moreover, the dimension of comparison in (7-b) and (8-b) is different. While in the case of arithmetical entities, what is relevant is their relative ordering on the number scale, the truth conditions in (8-b) concern the size of individual objects making up a plurality. For instance, the sentence would be false when comparing three onions and two pumpkins. Notice also that number agreement in (7-b) and (8-b) is different.

- (7) a. Hanna can count (up) to **three**.
 b. **Three** is bigger than two.
- (8) a. #Hanna can count (up) to **three** things.
 b. #**Three** things are bigger than two things.

Yet another property of abstract-counting numerals is that they do not give rise to scalar implicatures (Bultinck 2005). While the most natural way to understand (9) is that the addressee needs to give the speaker at least three things to pay their debt, (10) lacks an analogous lower bounded reading and can only get an *exact* interpretation.

- (9) You must give me **three** things to pay your debt. ✓AT LEAST
- (10) You must divide six by **three** to get two. #AT LEAST

Finally, only object-counting numerals allow for modification by comparative and superlative modifiers as well as so-called prequantifiers, e.g., the English universal quantifier *all* (cf. Corbett 1978). While the sentences in (11) are regular felicitous expressions, the examples with abstract-counting numerals in (12) are funny.

- (11) a. More than **three** cats live in the barn.
 b. At least **three** cats from the barn are crazy.
 c. All **three** cats who live in the barn are cute.
- (12) a. #More than **three** is a natural number.
 b. #At least **three** times two equals six.
 c. #All **three** is prime.

So far, we have seen that there is robust evidence that the conceptual distinction between abstract and object counting is encoded in natural language semantics and grammar. Given that all the examples discussed heretofore include the very same morphological form, an obvious question regarding the relationship between abstract and object counting arises.¹

2.2 Relating abstract and object counting

The contrasts discussed in the previous section call for an explanation of the dual life of cardinals such as English *three*. In order to account for the relationship between number concepts and pluralities, most mainstream approaches take the object-counting function of numerals as the basic one from which the abstract-counting meaning can be derived. For instance, Rothstein (2017) following Landman (2003) takes cardinal numerals to denote cardinal properties. As such they are intersective modifiers of type $\langle e, t \rangle$.² The abstract-counting meaning is then derived by a special shifting operation which applies to a cardinal property and yields a number-denoting expression of type n .

Another account assuming essential object-counting semantics for numerals is proposed by Ionin and Matushansky (2018). They posit that all cardinals are predicate modifiers (type $\langle \langle e, t \rangle, \langle e, t \rangle \rangle$) which provide the cardinality of a partition of a plural individual. In order to derive the abstract-counting meaning they propose a null nominalizing suffix. Its function is to yield the number corresponding to the object-counting denotation of a given cardinal.

Finally, Kennedy (2015) also argues that the basic meaning of numerals is object-counting. On his account, cardinals denote quantifiers over degrees (type $\langle \langle d, t \rangle, t \rangle$), e.g., *three* denotes a set of degree properties whose maximal value is 3. The abstract-counting meaning is then obtained as a result of the consequent application of the standard type-shifting operations BE and IOTA to the the degree-quantifier object-counting semantics.

Though in recent years treating abstract counting as derived seems to have become prevalent, there are also proposals deriving the object-counting meaning of numerals from a number-denoting core. Two examples of such accounts are Krifka (1995) and Hackl (2000). In Krifka's system the underlying meaning of cardinals amounts to the reference to a number concept. In addition, depending on a language this core meaning can be accompanied by an overt or covert object-counting element, e.g., a classifier. Likewise, Hackl starts with the abstract-counting meaning which can be then shifted to a GQ-style object-counting determiner.

What both families of theories briefly described above have in common is that they presume a relationship between the abstract- and object-counting component of numerals, an assumption justified by the identical form of numerals like *three* in both types of uses. How they account for the relationship in question is represented schematically in (13-a) and

¹An anonymous reviewer points out that in some cases, one can imagine contexts where the sentences that we mark as infelicitous can be used. While this is an interesting domain of inquiry (namely how to manipulate the context to make such sentences acceptable), this does not challenge our main claim. The claim is that there is a contrast between the two different uses of numerals; specifically, while one type of numerals is natural in the relevant examples, for the second type, one must work hard on the context to make them sound more natural.

²Actually, Rothstein (2017) distinguishes typically between lower numerals and multiplicands such as *hundred* but since this paper focuses only on basic cardinals such as *three*, we will ignore this intricacy.

(13-b), where ABS and OBJ stand for ‘abstract-counting’ and ‘object-counting’, respectively, and α and β are operations responsible for the shifts.

- (13) a. $\llbracket \text{three}_{\text{ABS}} \rrbracket = \llbracket [\alpha [\text{three}_{\text{OBJ}}]] \rrbracket$
 b. $\llbracket \text{three}_{\text{OBJ}} \rrbracket = \llbracket [\beta [\text{three}_{\text{ABS}}]] \rrbracket$

On the assumption that morphological marking reflects meaning composition, the two approaches make different predictions regarding morphological make-up of cardinal numerals. In particular, (13-b) predicts that across languages one should find numerals expressing α overtly. This means that cross-linguistically, abstract-counting numerals should exhibit a tendency to be morphologically more complex than object-counting numerals. On the other hand, (13-a) predicts the opposite, i.e., abstract-counting numerals should be morphologically more complex than object-counting numerals.

In the next two sections, we will explore cross-linguistic morphological marking patterns focussing on the marking of the two types of numerals. In Section 3, we present data that call into question the idea that the object-counting meaning of cardinals is basic, as in (13-a), while supporting the view offered in (13-b). In Section 4, we discuss examples indicating that the view in (13-a) is essentially correct, it is still too simplistic in that even abstract-counting numerals are not grammatically atomic, and break down into at least two smaller meaning components.

3 Simplex numerals

Though the English data discussed so far do not show this, languages do in fact make various morphological distinctions between abstract and object counting (Hurford 1998, 2001). In this section, we start exploring these patterns by looking at numerals that are monomorphemic in the abstract-counting function. We shall call them **SIMPLEX** numerals. Comparing them to their object-counting counterparts, we have identified three marking patterns, as provided in Table 1. In the table, A and B are variables representing morphemes.

	SYNCRETISM	STACKING	SUPPLETION
ABSTRACT COUNTING	A	A	A
OBJECT COUNTING	A	A+B	B

Table 1: Simplex numerals

In the **SIMPLEX SYNCRETIC** pattern, the abstract- and object-counting forms are identical and both consist of a single morpheme. On the other hand, the **SIMPLEX STACKING** pattern is characterized by object-counting numerals being morphologically more complex. In addition to the abstract-counting core, they contain an additional morpheme. Finally, in the **SIMPLEX SUPPLETIVE** pattern, abstract- and object-counting numerals have different and formally-unrelated monomorphemic shapes.

It is important to emphasize that the patterns as described in Table 1 above are used to classify individual numerals rather than entire languages. That is because various patterns can co-exist within a single language (cf. Bale and Coon 2014).

Notice that the patterns represented in Table 1 do not exhaust the logical possibilities of variation. In particular, what is missing are abstract-counting numerals that would be more complex than their object-counting counterparts. While such examples have also been reported, uncontroversial examples are very rare.³ In this paper we limit our focus to the patterns in Table 1, where particularly the **STACKING** pattern and the **SYNCRETIC** pattern are the two cross-linguistically most common ways of marking the two numeral types.

3.1 Syncretism

The first pattern we will investigate makes no overt distinction between abstract- and object-counting numerals. In other words, both functions are expressed by the same formal exponent. For instance, in English the prenominal modifier *three* is used to quantify over individuals in (14-a). In this use, it has the same shape as the name of a number concept, see (14-b). We call such numerals **SYNCRETIC**.

- (14) a. **three** apples
b. Two times **three** equals six.

From a typological perspective simplex syncretic numerals seem to be the most widespread type of numerals across languages. In (15)–(17), we illustrate this type by examples from Czech, Turkish and Basque, respectively.⁴

- (15) a. **pět** koček
five cats
'five cats'
b. **Tři** plus **pět** je osm.
three plus five is eight
'Three plus five is eight.' Czech
- (16) a. **üç** kız
three girl
'three girls'
b. **Yedi** eksi **üç** eşittir dört.
seven minus three equals four.
'Seven minus three equals four.' Turkish
- (17) a. **bi** lili
two flower
'two flowers'
b. **Hiru** bider **bi** sei dira.
three time two six are
'Three times two is six.' Basque

The syncretic pattern is not very revealing with respect to the relationship in question. In the next section, we examine cases where the abstract- and object-counting numerals differ

³One potential example is the German *ein-s* ~ *ein* (both 'one'); see Wągiel and Caha (2020) for a discussion and detailed analysis.

⁴We would like to thank Maia Duguine for her judgements and the discussion of the Basque data.

formally.

3.2 Stacking

The second pattern involves a formal asymmetry between the two types of numerals in question. In particular, object-counting numerals are morphologically more complex, i.e., include an additional morpheme compared to their abstract-counting counterparts. We shall refer to the formal relationship as **STACKING** and numerals that arise as a result of stacking will be called **AUGMENTED**.

Typically, the asymmetry is characteristic of classifier languages. In these languages, the additional morpheme used with object-counting numerals is often referred to as a classifier. For instance, in Mandarin, the numeral cannot appear bare when modifying an NP, see (18).

- (18) a. ***sān** qiú
 three ball
 Intended: ‘three balls’
 b. **sān** **gè** qiú
 three CLF ball
 ‘three balls’ Mandarin

On the other hand, the numeral has to be bare in an abstract-counting environment, see (19).⁵

- (19) a. jiǔ chùyǐ **sān** shì sān.
 nine divide.by three COP three
 ‘Nine divided by three is three.’
 b. #jiǔ **gè** chùyǐ **sān** **gè** shì sān **gè**.
 nine CLF divide.by three CLF COP three CLF
 Intended: ‘Nine divided by three is three.’ Mandarin

It should be noted that there is usually more to the semantics of classifiers than just allowing for numerical quantification. Classifiers often have also other functions, namely they introduce certain requirements regarding the kind, shape and size etc. of referents of the modified noun (e.g., Aikhenvald 2000). However, *gè* belongs in the class of the so-called general or default classifiers, which can be used to count virtually any type of object. Thus, rather than conveying some specific characteristic of the counted object, we understand Mandarin *gè* to be at its core a pure object-counting marker.⁶

The same pattern as in (18)–(19) is also attested in Vietnamese.⁷ As witnessed by the contrast in (20), the numeral *hai* ‘two’ must combine with an additional morpheme in an object-counting construction, where *cái* is a general classifier for inanimate objects.⁸

⁵We would like to thank Chang Liu for his judgements and the discussion of the Mandarin data.

⁶Note also that the closely related Dungan language (Sinitic) employs only one generic classifier *kə* for all nouns and a similar tendency is also present in many Northern Chinese dialects (Yue 2017, 114–115).

⁷We would like to thank Tue Trinh for his judgements and the discussion of the Vietnamese data.

⁸For a corpus-based study of the distribution of *cái*, see Pham and Kohnert (2008).

- (20) a. ***hai** bát
two bowl
Intended: ‘two bowls’
b. **hai cái** bát
two CLF bowl
‘two bowls’Vietnamese

On the other hand, in an abstract-counting environment like (21), the cardinal has to appear bare.

- (21) a. Bốn nhân **hai** bằng tám.
four times two equal-to eight
‘Four times two equals eight.’
b. #Bốn cái nhân **hai cái** bằng tám cái.
four CLF times two CLF equal-to eight CLF
Intended: ‘Four times two equals eight.’Vietnamese

Yet another example comes from Adang (Alor-Pantar), a Papuan language with a developed classifier system. Similar to Mandarin and Vietnamese, Adang object-counting numerals are bimorphemic, as demonstrated in (22), where *pa?* is a general classifier (Robinson and Haan 2014, 250). On the other hand, abstract-counting cardinals are monomorphemic (Klamer et al. 2014, 354).

- (22) ʔaburiŋ **pa?** ut
arrow CLF four
‘four arrows’Adang
(23) air nu ‘aba’ang ʔo-alu **iwi**hing
ten one divide POSS-two five
‘Ten divided by two is five.’Adang

Augmented numerals occur also in Indo-European. For instance, in Bhojpuri (Indo-Aryan) the morphemes *go* and *tho* function as general classifiers that can be used to quantify over an object of any type, size or shape. An example of an object-counting use is given in (24), where the bimorphemic form consisting of the numeral root and *go* is used as a prenominal modifier (Barz and Diller 1985, 166; adapted). However, in an abstract-counting context neither *go* nor *tho* is employed, see (25) (Barz and Diller 1985, 165).

- (24) **sāt go** nokar
seven CLF servants
‘seven servants’Bhojpuri
(25) das ā **pāc** hai panarā, nā?
ten and five are fifteen no
‘Ten and five make fifteen, don’t they?’Bhojpuri

The final example to be discussed here comes from Yoruba (Niger-Kongo), which is a non-classifier language. At least in some varieties of Yoruba, the distinction between abstract-

counting and object-counting numerals is marked by the prefix *m-*.⁹ As demonstrated in (26) and (27), only prefixed forms can be used as nominal modifiers whereas only bare numerals can express abstract counting.¹⁰

- (26) a. *ìwé **ẹ̀ta**
book three
Intended: ‘three books’
b. ìwé **m-ẹ̀ta**
book PRFX-three
‘three books’ Yoruba
- (27) a. ẹ̀jì pelu **ẹ̀ta** je àrún
two with three earn five
‘Two plus three equals five.’ Yoruba
b. #m-ẹ̀jì pelu **m-ẹ̀ta** je m-árún
PRFX-two with PRFX-three earn PRFX-five
Intended: ‘Two plus three equals five.’

Cross-linguistically, the asymmetry discussed in this section is a relatively frequent phenomenon. Other languages with simplex numerals exhibiting the stacking pattern reported in the literature include, e.g., Thai (Hundius and Kölver 1983), Japanese (Sudo 2016), Assamese (Indo-Aryan) (Kalita 2011, 184) and Teiwa (Alor-Pantar) (Klamer 2010, 144–145; Klamer et al. 2014, 351). We interpret this phenomenon as evidence for the view that the abstract-counting meaning is basic and the object-counting meaning is derived from it, recall (13-b) in Section 2.2.

3.3 Suppletion

Finally, the third marking pattern to be discussed here concerns SUPPLETIVE numerals. In such a case, for a given number there are two simplex morphologically unrelated forms one of which expresses abstract counting whereas the other is used in the object-counting function. Admittedly, this pattern is very rare and it is challenging to provide an uncontroversial instance of a suppletive abstract/object-counting pair. Nonetheless, Hurford (1998) gives the example of numeral suppletivism in Maltese in which there are two morphologically unrelated forms corresponding to the number 2, namely *tnejn* and *żewġ* (both ‘two’). As evidenced by the contrast in (28), only *żewġ* can convey the object-counting meaning as a nominal modifier (Borg 1974, 293). On the other hand, only *tnejn* is possible in an abstract-counting environment such as (29) (Borg 1987, 62).¹¹

⁹The occurrence of *m-* is also accompanied by a prosodic change, namely the low tone is replaced by the high tone (Ajiboye 2016).

¹⁰Ajiboye (2005, 235) states that both bare and prefixed forms can occur in some object-counting environments, but Ajiboye (2016, 1, fn 3) notes that nowadays many speakers prefer prefixed forms in all object-counting contexts. In any case, our informant accepts only *m-* numerals for object counting. We would like to thank Mary Chimaobi Amaechi for her judgements and the discussion of the Yoruba data.

¹¹It should be noticed that Borg (1974) proposes that the distinction between *tnejn* and *żewġ* could be accounted for syntactically. Specifically, he proposes that *tnejn* is an independent form used in the absence of a noun, whereas *żewġ* can only be used when preceding an NP. For the sake of space, we will not dive into

- (28) a. ***tnejn** nisa
two_{ABS} women
Intended: ‘two women’
b. **żewġ** nisa
two_{OBJ} women
‘two women’Maltese
- (29) a. **Tnejn** u **tnejn** jagħmlu erbgħa.
two_{ABS} and two_{ABS} they.make four
‘Two and two make four.’
b. ***Żewġ** u **żewġ** jagħmlu erbgħa.
two_{OBJ} and two_{OBJ} they.make four
Intended: ‘Two and two make four.’Maltese

As already mentioned above, the suppletive pattern is very rare and if identified, it typically applies to only one numeral in a language. However, according to the literature it is not restricted to Maltese ‘two’. Other examples of simplex suppletive numerals reported so far include, e.g., the cardinals *ū* and *hleh* (both ‘one’) in Palaung (Austroasiatic) (Greenberg 1978) as well as the Iban (Eastern Ijaw) numerals *gberę* and *ngię* (both ‘one’) and *oyi* and *atie* (both ‘ten’) (Obikudo 2016). In Section 7, we will discuss in detail an additional example from colloquial Mandarin (He 2015).

The final note on suppletion we need to make here is that STACKING and SUPPLETION may in fact combine. As a case in point, consider the Mandarin numeral ‘two.’ The numeral has the form *èr* for abstract counting, see (30-a) (from Po-Ching and Rimmington 2015, 26). In (30-b), we see that the object-counting use features a suppletive root *liǎng* ‘two,’ crucially accompanied by the general classifier *gè* (He 2015, 198). It is ungrammatical to use the abstract-counting *èr* in such phrases, see (30-c) (He 2015, 198). In sum, the conclusion is that in the example (30-b), stacking and suppletion combine.

- (30) a. yī jiā yī děngyú **èr**
one add one equals two
‘One plus one is two.’
b. **liǎng** **gè** xuéshēng
two CLF student
‘two students’
c. ***èr** **gè** xuéshēng
two CFL student
intended: ‘two students’Mandarin

3.4 A side note on compound numerals

In our investigation, we often found a relation (suggestive, even though possibly an imperfect one) between the shape of abstract-counting numerals and the shape of numerals within

the details of Borg’s and Hurford’s accounts and we simply acknowledge that an alternative explanation of the pattern is possible. We would like to thank Albert J. Borg for his judgments and the discussion of the Maltese data.

compound numerals. We use the term compound numerals to refer to numerals like *two hundred five* that arise by combining two or more basic numeral roots.

In order to see the parallel between complex numerals and abstract counting, recall the Mandarin example (19) repeated in (31). The example shows that classifiers are absent in mathematical statements.

- (31) a. jiǔ chùyǐ sān shì sān.
 nine divide.by three COP three
 ‘Nine divided by three is three.’
 b. #jiǔ gè chùyǐ sān gè shì sān gè.
 nine CLF divide.by three CLF COP three CLF
 Intended: ‘Nine divided by three is three.’ Mandarin

As pointed out by He (2015, 195), the classifier is also (unsurprisingly) missing in arithmetical statements like (32), where the property of being a prime number (which, recall, is a property of abstract numbers) is predicated of a compound numeral composed of three other numerals (‘two,’ ‘ten’ and ‘three’).

- (32) èr shì sān shì sùshù.
 two ten three be prime number
 ‘Twenty three is a prime number.’ Mandarin

These three numerals jointly specify (by multiplication and addition) the number 23, which is, indeed, a prime number. What is interesting for us is that in order to arrive at the meaning of the compound numeral, the arithmetical operations such as multiplication and addition must be postulated among the numeral roots, namely $[23 = 2 \times 10 + 3]$. Such operations are, in turn, defined over abstract-counting numerals, and we are therefore not surprised that there is no classifier in between any two numerals in (32).

Interestingly, the absence of classifiers in between numerals is also observed in object-counting contexts, see (33) from He (2015, 190).

- (33) [[èr shí sān] wàn] gè xuéshēng
 two ten three ten.thousand CLF student
 ‘two hundred thirty thousand students’ Mandarin

He (2015) gives multiple arguments in favor of the proposal that the four basic numerals in (33) (i.e., *èr*, *shí*, *sān* and *wàn*) form a constituent to the exclusion of the classifier. According to He, the numerals are first joined together compositionally by a series of arithmetical operations $[(2 \times 10 + 3) \times 10,000]$ and they jointly form a complex cardinal ‘two hundred thirty thousand.’ This internally complex compound cardinal is then used to quantify over the denotation of the noun, at which point the classifier becomes indispensable, turning the abstract numeral into an object-counting numeral (cf. Cinque 2020). In sum, the absence of classifiers inside complex numerals suggests that individual numerals inside compounds are similar (both semantically and morphologically) to abstract-counting numerals.

This view is supported by the fact that the form of the initial numeral ‘two’ in (33) is *èr*. Recall from (30) – repeated in (34) – that *èr* ‘two’ is a dedicated abstract-counting form of ‘two;’ see (34-a). Therefore, the appearance of *èr* in an object-counting compound numeral

in (33) indicates that inside such numerals we find the abstract-counting structure.

- (34) a. yī jiā yī děngyú èr
 one add one equals two
 ‘One plus one is two.’
 b. *èr gè xuéshēng
 two CLF student
 intended: ‘two students’
 c. liǎng gè xuéshēng
 two CLF student
 ‘two students’

Mandarin

He (2015, 198-9) points out that the abstract-counting form *èr* is also found in additive numerals. This is interesting, because the sequence *èr*+CLF is generally ungrammatical, recall (34-b). However, in an additive numeral like ‘twelve,’ the form *èr* must be used in object-counting contexts, see (35-a). The form *liǎng* ‘two’ (which is found before the classifier in (34-c)) is impossible in compound numerals, see (35-b) (He 2015, 198). Once again, the fact that *èr* is used in compound numerals suggests that the internal structure of such numerals is similar to abstract-counting structures.

- (35) a. [wǔ shí èr] gè xuéshēng
 five ten two CLF student
 ‘fifty two students’
 b. *[wǔ shí liǎng] gè xuéshēng
 five ten two CLF student
 intended: ‘fifty two students’

Mandarin

The reason why these facts are relevant to note is because abstract-counting contexts (arithmetical examples) are often missing in grammars. This limitation may be overcome in a number of cases by investigating compound numerals, which (in a number of interesting cases) seem to mirror the structure of the Mandarin example in (35-a).

Consider, for example, the examples found in Upper Necaxa Totonac, as discussed in Beck (2004, 26-7). In Upper Necaxa Totonac, numerals in object-counting functions precede the noun and they are themselves preceded by a classifier, see (36-a) for the numeral ‘one’ with a classifier for bunches. In (36-b), we see the numeral ‘ten’ with a general classifier preceding it. This numeral is given here in order to see that the numeral ‘eleven’ in (36-c) is composed by juxtaposing the roots of the two numerals in (36-a,b).

- (36) a. kiłmak-**tín** más-nĩ séʔnu
 CLF-one rot-DVB banana
 ‘a bunch of [rotten] bananas’
 b. aʔ-**káux**
 CLF-ten
 ‘ten’
 c. aʔ-**káux-tín**
 CLF-ten-one
 ‘eleven’

Upper Necaxa Totonac

The classifier is found only once in the compound numeral, suggesting that (on analogy with Mandarin) the numeral ‘eleven’ is formed first as an abstract-counting numeral [10+1]. Only after the compound numeral is formed, the classifier attaches to it, yielding an object-counting numeral. If correct, this suggests that Upper Necaxa Totonac provides us with yet another example of the stacking pattern.

The final example of this type we shall mention here comes from the Nilotic language Luwo (Storch 2014, 271). In this language, the numerals ‘one’ to ‘five’ are basic (not compound) and we list them in the first column in Table 2. It is interesting to note that they all start with an initial *á*, which is placed in italics and separated as an independent morpheme. The residue of the numeral is bold. The same strategy is applied to the numeral ‘ten’ at the bottom of the second column, which is also basic (not compound).

NUMBER	NUMERAL	NUMBER	NUMERAL	NUMBER	NUMERAL
1	<i>á</i> - cíèl ó	6	<i>á</i> -buc bí cíèl	20	dháànhò àdòuònò
2	<i>á</i> - ríòw	7	<i>á</i> -buc bí ríòw	40	jé- ríòw
3	<i>á</i> - dák	8	<i>á</i> -buc bí dák	60	jé- dák
4	<i>á</i> - ɲwèèn	9	<i>á</i> -buc bí ɲwèèn	80	jé- ɲwèèn
5	<i>á</i> -buc	10	<i>á</i> - pààr	100	jé-buc

Table 2: Luwo numerals

Let us now turn to the numerals ‘six’ to ‘nine.’ These are contained in the second column. We can see that they are always composed of the numeral ‘five,’ followed by *bí* (glossed as ‘plus’ in Storch 2014) and an appropriate numeral from the basic set. What is most relevant for our purpose is that the second numeral (after *bí*) lacks the initial *á*-. This suggests a structure where the *á* is a separate morpheme that attaches to the compound numeral as in: *á*-[5+1], *á*-[5+2], etc. This again suggests that a compound numeral is formed first (through addition), and only after the compound numeral is created, we attach a ‘general classifier’ *á*- in order to turn the abstract numeral into an object-counting numeral. As in Upper Necaxa Totonac, this suggests that Luwo numerals belong in the stacking type, since in the general classifier is missing in compound numerals, which resemble abstract-counting structures.

The same conclusion is suggested by looking at the multiplicative numerals ‘forty,’ ‘sixty,’ ‘eighty’ and ‘hundred’ in the last column. These numerals contain the base *jé*- glossed as ‘person’ in Storch (2014), but obviously standing in for ‘twenty.’ This invariant base is never preceded by *á*- and it is therefore classified as a syncretic numeral in our typology. What is relevant is that when *jé*- is multiplied, it is followed by the *á*-less numeral roots of the numerals ‘two’ to ‘five.’ The absence of the general classifier *á*- once again suggests that the numerals 1-5 represent a stacking pattern that obtains between *á*-less abstract-counting numerals and object-counting numerals (the latter include the *á*-).¹²

¹²Interestingly, the addition of the numerals 1-9 to 10 is done in a different way, where both numerals contain the ‘classifier’ *á*, as in (i).

(i) *á*-pààr ɲwóɲ=é *á*-cíèló
 CLF-ten increase:tr=3sg CLF-one
 ‘eleven’ lit. ‘ten increases one’

3.5 Summary

The data explored in this section (the STACKING pattern discussed in subsection 3.2 in particular) suggest that across languages, the abstract-counting component of numerals is more primitive than the object-counting one. This is because of the fact that in the STACKING cases, the object-counting numeral morphologically contains the abstract-counting numeral. In the next section, we explore cross-linguistic evidence suggesting that even abstract-counting cardinals involve more structure than typically assumed.

4 Complex numerals

In this section, we turn our attention to languages in which abstract-counting numerals are morphologically complex in that they contain two morphemes. We shall call such abstract-counting numerals COMPLEX numerals. The patterns we found in such languages are summarized in Table 3 (A, B and C again represent separate morphemes).

	SYNCRETISM	STACKING	SUPPLETION
ABSTRACT COUNTING	A+B	A+B	A+B
OBJECT COUNTING	A+B	A+B+C	A+C

Table 3: Complex numerals

The types of relations in Table 3 are identical to what we have observed in simplex cardinals, recall Table 1; the only difference is that the first row of the table 3 contains a numeral that is bimorphemic (rather than non-decomposable). For example, in the complex syncretic pattern, there is (again) no formal distinction between abstract- and object-counting numerals. The only difference to what we have already seen is that the syncretic numerals in Table 3 consist of two separate morphemes. Similarly, the complex stacking pattern again reveals an asymmetry between abstract- and object-counting numerals, only this time, the former are already bimorphemic. As a result, the object-counting form contains an additional third morpheme. Finally, according to the complex suppletive pattern, both abstract- and object-counting cardinals share the same core but differ in that each employs a different (morphologically unrelated) affix.

The data thus support the basic analytic claim (namely that object-counting numerals are derived from abstract-counting numerals), but lead us to revise the hypothesis that abstract-counting numerals are structurally simplex. Specifically, if each morpheme in the abstract-

This suggests that languages probably allow for (at least) two different ways of creating additive numerals. The second option, recently argued for in Ionin and Matushansky (2006), is that additive numerals originate as complex coordinations of the sort shown in (ii-a), where the first noun is unpronounced (possibly due to Right Node Raising). This structure could be the source for (i), while the examples discussed in the main text would arise by employing the structure (ii-b). It must be noted, though, that the structure in (ii-a) is only available for the formation of object-counting numerals.

- (ii) a. [CLF-NUMERAL \mathbb{N}] & [CLF-NUMERAL N]
b. CLF-[NUMERAL & NUMERAL] N

counting pattern expresses an independent component of meaning, we have to decompose abstract-counting numerals into at least two ingredients.

We also add here again that the patterns to be discussed pertain to individual numerals rather than whole languages, since a given language can have more than one type of cardinals. A description of an entire numeral system of a language will result from providing a full classification of its numerals into various marking patterns.

4.1 Syncretism

Let us begin the empirical discussion from COMPLEX SYNCRETIC numerals. A good example is Hawaiian, where the cardinals 1–9 always consist of a root and the prefix ‘e-’.¹³ As witnessed in Table 4 (based on Elbert and Pukui 1979, 158–160), the interpretation of ‘e-’ as an affix is corroborated by the fact that it is replaced by another morpheme in distributive numerals.¹⁴

NUMBER	CARDINAL	DISTRIBUTIVE
2	‘e-lua	pā-lua
3	‘e-kolu	pā-kolu
4	‘e-hā	pā-hā
5	‘e-lima	pā-lima
6	‘e-ono	pā-ono

Table 4: Numerals in Hawaiian

An example of an object-counting use of the numeral ‘e-lua’ ‘two’ in Hawaiian is provided in (37) where the prefixed form precedes the noun (Elbert and Pukui 1979, 159).

- (37) ‘e-lua i‘a
PRFX-two fish
‘two fish’ Hawaiian

Interestingly, the very same shape is also used in mathematical discourse to express the abstract-counting function, as demonstrated in (38) (Pukui and Elbert 1986, 501; adapted). These data show that unlike, e.g., in Mandarin, Hawaiian cardinals are morphologically complex in both functions under consideration.

- (38) ‘e-lua ā me ‘e-lua, ‘e-hā
PRFX-two and with PRFX-two PRFX-four
‘Two plus two is four.’ Hawaiian

Another example of complex syncretic numerals comes from Sanzhi Dargwa (Northeast Caucasian). In this language, all basic cardinals except for *ca* ‘one’ are morphologically complex. They consist of a root and a special derivational suffix *-(j)al*, see Table 5 (based on Forker

¹³The prefix can be sometimes realized as ‘a- and it can be dropped only in a counting sequence as in (‘e-)kahi, (‘e-)lua, (‘e-)kolu... (‘one, two, three...’) (Elbert and Pukui 1979, 158).

¹⁴We follow here Elbert & Pukui (1979) in using the label ‘distributive’ though at least some of the uses of numerals prefixed with *pā-* are clearly multiplicative.

2020, 130–135).¹⁵ The affixal status of *-(j)al* is proved by the structure of ordinal and multiplicative numerals in which *-(j)al* is replaced by another element.

NUMBER	CARDINAL	ORDINAL	MULTIPLICATIVE
2	k’w -el	k’wi -ʔibil	k’wi -jna
3	ʔa^fb -al	ʔa^fb -ʔibil	ʔa^fb -jna
4	aɸ^w -al	aɸ -ʔubil	aɸ^w -na
5	xu -jal	xu -ʔibil	xu -jna
6	urekɿ -al	urek -ʔibil	urek -na

Table 5: Numerals in Sanzhi Dargwa

The bimorphemic forms in the left compartment of Table 5 are used as both object- and abstract-counting numerals. For instance, in (39), the cardinal *xu-jal* ‘five’ functions as a prenominal modifier (Forker 2020, 131; adapted).

- (39) **xu-jal** rurs:i
 five-SFX girl
 ‘five daughters’ Sanzhi Dargwa

The very same form is also used in abstract-counting environments such as (40) (Forker 2020, 137; adapted).

- (40) k’wel-le ʔa^fb-al či-ka-b-ix-ar b-irχ^w-u arg-u **xu-jal**
 two-LOC three-SFX SPR-down-N-throw-COND N-become-PRS go.IPFV-PRS five-SFX
 ‘Two plus three equals five.’ Sanzhi Dargwa

Having introduced the pattern of COMPLEX SYNCRETIC cardinals, we now turn to a language with COMPLEX numerals that exhibit the STACKING pattern.¹⁶ This language behaves similarly to, e.g., Mandarin or Yoruba, but features abstract-counting numerals that are morphologically complex.

4.2 Stacking

The relevant pattern is found in Vera’a (Vanuatu). In this language, the cardinals 1–5 are morphologically complex, containing a root preceded by the prefix *vō-/ve-*, see Table 6 (based on Schnell 2011, 73–74).¹⁷

The status of *vō-/ve-* as a separate morpheme is evidenced by the fact that it is replaced by the prefix *vag-* in multiplicative numerals. Analogous prefixes on basic cardinals are attested in other Vanuatu languages as well, e.g., in Vurës (Malau 2016, 126).

¹⁵The suffix has the shape *-al* after consonants, *-jal* after vowels and *-el* in the numeral ‘two’, apparently as a result of contraction (Forker 2020, 129–130).

¹⁶The complex syncretic pattern is also found in Japhug (Sino-Tibetan, Jacques 2018).

¹⁷The allomorph *ve-* appears only in the numeral ‘five’ (Schnell 2011, 73). Cardinals higher than ‘five’ are compound expressions with the structure *livi-X* ‘five plus X’, where X is substituted by the relevant unprefixed numeral; e.g., *livi-ru(ō)* ‘seven.’ According to our classification these would fall into the simplex category.

NUMBER	CARDINAL	MULTIPLICATIVE
1	vō- wal	vag- wal
2	vō- ru (ō)	vag- ru (ō)
3	vō-' ōl	vag-' ōl
4	vō- ve'	vag- ve'
5	ve- limē	vag- limē

Table 6: Numerals in Vera'a

In abstract-counting contexts like the one exemplified in (41) (from Schnell 2011, 83), we find the numeral in the same form as given in the table. This pattern is similar to the ones seen in the previous section, where abstract-counting numerals were morphologically complex.

- (41) vēvē-gi ne lukun ēn naw, din ēn **vō-'**ōl****...
mother-3SG TAM count ART wave reach ART NBR-three
'Then his mother counted the waves reaching (the number) three...' Vera'a

What is different about Vera'a is that in NPs, i.e., in the object-counting function, the numerals appear with an additional morpheme *ne* referred to as a 'ligature' by (Schnell, 2011, 74), see (42).

- (42) ēn woqe'enge **ne vō-ru**
ART tree LIG NBR-two
'two trees' Vera'a

It is interesting to note that according to Schnell's description, the ligature is specific to numerals, i.e., it is not a general linking morpheme. In sum, Vera'a instantiates a pattern similar to Mandarin, because its object-counting numerals stack an additional marker on top of the abstract-counting numeral. The new thing about Vera'a is that the abstract numeral is already complex, so we have a sequence of three morphemes in Vera'a.¹⁸

The significance of the Vera'a data is that it suggests the abstract- and object-function consist of two and three semantics components, respectively, each of which can be expounded by a separate morpheme. Although instances of such a strongly agglutinative marking are scarce, the strength of the evidence from Vera'a indicates two things. First, morphologically complex abstract-counting cardinals suggest that we need a complex structure for them. And further, the data shows that even morphologically complex abstract numerals may be even further augmented in their object-counting function.

¹⁸Schnell notes that the ligature is not always found with the numeral in the object-counting function, describing it as optional in this context. Concerning the ligature-free uses of the numeral, Schnell (2011, 74) says "that often the combination of noun and bare numeral is a fixed lexicalized expression." In our account, we focus only on the pattern where the ligature is present inside NPs, and we leave an account of the optionality for future work.

4.3 Suppletion

The final pattern that we would like to discuss is attested in a subset of languages that would be classified as obligatory-classifier languages. In such languages, the numeral in the object-counting function is always accompanied by one of a potentially large set of classifiers. We depict this schematically in (43-a). This is similar to Mandarin and other languages discussed in Section 3.2.

- (43) A mixed pattern
- a. Object-counting: NUMERAL + CLF1/CLF2/CLF3/...
 - b. Abstract counting: NUMERAL + CLF1

However, unlike in Mandarin, the languages that we shall discuss in this section retain the classifier in the abstract-counting function, as depicted in (43-b). The particular classifier tends to be one of the set of object-counting classifiers. Following the tradition, we will call it the general classifier.

If we analyze the pattern in (43), we realize that such languages represent a mixture of two ‘pure’ systems. On the one hand, such numerals have something in common with the COMPLEX SYNCRETIC numerals discussed in Section 4.1. The similarity consists in that the particular form consisting of a NUMERAL + CLF 1 (the general classifier) has both an object-counting use and an abstract-counting use.

- (44) A pure COMPLEX SYNCRETIC pattern
- a. Object-counting: NUMERAL + CLF1
 - b. Abstract counting: NUMERAL + CLF1

However, the language type in (43) is different from the pure COMPLEX SYNCRETIC pattern (as exemplified by the languages discussed in Section 4.1) in that the abstract-counting classifier is not always found in the object-counting function, and may be replaced by other (more specific) classifiers. If we only focused on these other classifiers, we would get a pattern where the abstract-counting classifier is replaced by an unrelated (i.e., suppletive) object-counting classifier, as in (45).

- (45) A pure COMPLEX SUPPLETIVE pattern
- a. Object-counting: NUMERAL + CLF2/CLF3/...
 - b. Abstract counting: NUMERAL + CLF1

Based on these considerations, we shall treat here the mixed type separately from the syncretic type, and, in fact, as closely related to the suppletive type. However, we admit that these languages do not represent the pure suppletive type depicted (45). Such a pure suppletive type of language was not found among the languages we looked at.

As the first example of the mixed/suppletive type, consider Shuhi (Qiangic, Sino-Tibetan). This language has a relatively rich inventory of classifiers. They are obligatory in numeral phrases and their choice depends on the modified noun. One of them is *ko*³⁵, which serves as the general classifier, see (46-a) (Qi and He 2019, 65).¹⁹ The example (46-b) contains a

¹⁹Numbers in the superscript represent tones.

different classifier, namely *tshu*⁵⁵.

- (46) a. *ze*³¹*mi*³¹ *ne*³³-**ko**³⁵
 child two-CLF
 ‘two children’
 b. *la*³³*re*⁵⁵ *dzi*³³-**tshu**⁵⁵
 towel one-CLF
 ‘one towel’

Shuhi

Importantly, unlike languages such as Mandarin or Yoruba, Shuhi numerals require the general classifier *ko*³⁵ also in abstract-counting contexts such as (47) (Qi and He 2019, 69).

- (47) *dzi*³³-*ko*³⁵-*re*³³ *dzi*³³-*ko*³⁵-*fi*³³ *me*³³-*ba*³³-*le*⁵⁵ *ne*³³-**ko**³⁵ *le*³³-*zi*³³-*dzi*³³.
 one-CLF-ABL one-CLF-LOC DIR-add-AUX two-CLF DIR-become-DUR
 ‘One plus one is two.’

Shuhi

A similar set of facts is found in Mokilese (Austronesian). In (48), we can see the examples of two classifiers (Harrison 2019, 91). The general classifier *-w* is seen in (48-b). The pair nicely illustrates the general nature of *-w*.

- (48) a. *adroau riah*-**kij**
 egg two-CLF
 ‘two pieces of egg’
 b. *adroau ria*-**w**
 egg two-CLF
 ‘two eggs’

Mokilese

The example in (49) again contrasts the general classifier *-w* in (49-a) and the classifier *-pas* for long thin objects, see (49-b). In (49-c), we see the animate classifier *men*.

- (49) a. *wus ria*-**w**
 banana two-CLF
 ‘two bananas’
 b. *wus rah*-**pas**
 banana two-CLF
 ‘two banana trees’
 c. *jeri roah*-**men**
 children two-CLF
 ‘two children’

Mokilese

The abstract-counting context is shown in (50). We can see that the numeral *walu*- ‘eight’ has the general classifier *-w* after the numeral ‘eight’ (Harrison 2019, 94).

- (50) *Ria-pak pah-w walu-w*.
 two-times four-CLF eight-CLF
 ‘Two times four is eight’.

Mokilese

A similar system is found in Abkhaz (Northwest Caucasian). In the morphological system of this language, nouns are grammatically classified as human or non-human, with human nouns

including the classes of masculine and feminine nouns (Chirikba 2003, 24–25). Furthermore, the cardinals 2–10 can take two forms, as demonstrated in Table 7 (based on Chirikba 2003, 34–35 and Hewitt 2010, 33) each of which is clearly morphologically complex.²⁰

NUMBER	ABSTRACT/NON-HUMAN	OBJECT HUMAN
4	pš' -ba	pš' -j°ó(k')
5	x° -ba	x° -j°ó(k')
6	f -ba	f -j°ó(k')
7	bəž' -ba	bəž' -j°ó(k')
8	aa -bá	aa -j°ó(k')

Table 7: Cardinals in Abkhaz

Forms with the suffix *-ba* are object-counting cardinals used to quantify over referents of non-human nouns, see (51-a). On the other hand, numerals with the suffix *-j°ó(k')* are dedicated for counting human individuals, see (51-b) (Hewitt 2010, 35; transliterated and adapted).

- (51) a. a-c°a-k°a **x°-ba**
 ART-apple-PL five-NH
 ‘(the) five apples’
 b. a-č°k°ónc°a **x°-j°ó**
 ART-boy-PL five-N
 ‘(the) five boys’

Abkhaz

Both types are morphologically marked, i.e., there is no structural asymmetry between the two forms. However, despite this fact, only non-human numerals can express the abstract-counting meaning, as evidenced by the contrast in arithmetical statements such as (52).²¹

- (52) a. j°-ba jacəwc°ar x-pa jəq°alojt° **x°-ba**
 two-NH if.you.add three-NH it.becomes five-NH
 ‘Two plus three equals five.’
 b. #j°ó-ž°a jacəwc°ar x-j°ó jəq°alojt° **x°-j°ó**
 two-H if.you.add three-H it.becomes five-H
 Intended: ‘Two plus three equals five.’

Abkhaz

Admittedly, none of the patterns discussed here is an unequivocal instance of abstract/object-counting suppletivism. Nevertheless, based on the reasoning presented at the start of this section, we treat this mixed type as a relevant case of suppletivism, because this is what one part of the system amounts to. The reason why it is important to capture the fact that the general classifier is replaced by other classifiers is because it is not necessary for it to be so. For example, in the Mayan language Chuj, the more specific classifier (called nominal classifier) may (optionally) appear in addition to the numeral classifier. For example, in (53),

²⁰The numeral ‘two’ takes the suffix *-ž°a* instead of *-j°ó(k')* and *-ba* has the form *-bá* in the numeral ‘eight’ and *-pa* in ‘three’ as a result of phonetic assimilation (Chirikba 2003, 34). In multiplicatives, *-ba/-j°ó(k')* are replaced by the suffix *-nt°*.

²¹We would like to thank Viacheslav Chirikba for the discussion of the Abkhaz data.

the animate numeral classifier *wanh* is followed by the nominal classifier *nok* ‘animal’ (Royer 2017, 33).

- (53) Ay chab'-**wanh nok**' tz'i'.
 exist two-NUM.CL NOM.CL dog
 ‘There are two dogs.’

Chuj

While we do not wish to suggest that Chuj represents a stacking pattern (there is much to be understood about the second – so-called nominal – classifier), we do think that the possibility of a system like the one in Chuj highlights the need to explain why the general classifier does not combine with other classifiers. And this is an essence of what we label as the COMPLEX SUPPLETIVE pattern.

Before we propose how to account for the morphosemantic variation in cardinal numerals discussed so far, let us summarize our findings.

5 Data summary

In the two previous sections, we have examined six types of meaning/form correspondences across numerals in various genetically and typologically distinct languages. As one can see in Table 8, the cross-linguistic variation we have explored reduces to the interplay of two factors. The first one is the morphological relationship between forms expressing the abstract- and object-counting meaning, i.e., syncretism, stacking and suppletion. The second one concerns the shape of the abstract-counting form, i.e., simplex vs. complex. Importantly, the patterns in Table 8 are numeral-specific rather than language-specific as different numerals in a single language can fall into various types.

	TYPE	LANGUAGE	NUMBER	ABSTRACT	OBJECT
SIMPLEX	SYNCRETISM	English	3	three	three
	STACKING	Mandarin	3	sān	sān-gè
	SUPPLETION	Maltese	2	tnejn	żewġ
COMPLEX	SYNCRETISM	Shanzi Dargwa	5	xu-jal	xu-jal
	STACKING	Vera'a	2	vō-ru(ō)	ne-vō-ru(ō)
	SUPPLETION	Mokilese	4	pah-w	pah-men

Table 8: Data Summary

In the simplex patterns, abstract-counting forms are monomorphemic, whereas in the complex patterns they are bimorphemic. Syncretic numerals do not differentiate morphologically between the abstract- and object-counting form. On the other hand, in the stacked and suppletive pattern, the two functions are expressed by different exponents. In the former case, object-counting numerals employ an additional morpheme compared to their abstract-counting counterparts, whereas suppletive numerals are not distinguished by their structural complexity but rather by the fact that they consist of morphologically unrelated forms.

The patterns summarized in Table 8 on the one hand suggest that the abstract-counting meaning of numerals is more basic than the object-counting meaning. On the other hand,

the existence of complex abstract numerals calls into question a widespread assumption that basic cardinal numerals are simplex expressions. If we adopt the assumption that morphology expresses some meaning, it follows from the existence of the complex stacking pattern that there are at least two, respectively three semantic components encoded by such numerals.

What we perceive to be a single thread running through the patterns is that whenever there is a difference in morphological complexity among object- and abstract-counting numerals, object-counting numerals are more complex. This apparently close-to universal asymmetry suggests to us that it makes sense to propose a unified theory where all numerals are characterized by the same ingredients, such that at the level of meaning, object-counting numerals are more complex than abstract-counting numerals. The remaining points of variation can be accounted for within a system of morphological realization.

6 Structures

In order to explain the asymmetry in marking patterns summarized in Table 8, we postulate a set of primitive semantic features that abstract- and object-counting numerals are composed of. We assume that those semantic primitives are cross-linguistically invariant and can feed into a unified morpho-semantic system. As a result, we will be able to compositionally derive the abstract/object counting distinction in a cross-linguistically uniform fashion, with variation limited to morphosyntactic realization. However, before we turn to the technical details of our analysis, let us spell out our assumptions regarding the nature of cardinals and the role of classifiers.

6.1 Key intuition

The core intuition behind our proposal is that numerals are at their core scalar entities. Specifically, we assume that each numeral is associated with an interval on the number scale. Assuming that the number scale has a non-arbitrary starting point corresponding to 0 and that each numeral designates a different upper bound, such intervals can be viewed as ordered in terms of their length.

It seems to us that there are at least three other strands of research that share a similar intuition. The first one concerns an attempt to explain the meaning of spatial and directional numeral modifiers as in *above three* and *up to three* (Nouwen 2016). Furthermore, in a number of approaches to degree semantics, degrees are taken to refer to intervals rather than to points on a scale (e.g., Seuren 1984; Kennedy 2001; Schwarzschild and Wilkinson 2002). Finally, the recent analysis of morphological marking patterns in comparative formation suggests a similar direction (Vanden Wyngaerd et al. 2020).²²

²²The idea that numerals are at their core associated to an interval is compatible with them having also other uses, since the interval can be mapped onto various types of objects, e.g., its extreme point. We shall return to this later.

6.2 The role of classifiers

The standard assumption regarding the role of numeral classifiers in languages such as Mandarin concerns the semantics of nouns (e.g., Chierchia 1998; Borer 2005; Rothstein 2010; Scontras 2013). That is because according to the received view the denotations of nouns in classifier languages are mass-like which makes them incompatible with the meaning of numerals. Thus, classifiers are required in order to compensate this semantic ‘deficit’ and turn the denotations of nouns into countable ones.

In this paper, however, we embrace an alternative view on the role of classifiers in numeral phrases (Krifka 1995, Bale and Coon 2014, Sudo 2016). One of the problems the received view encounters concerns languages such as Mi’gmaq (Algonquian) and Ch’ol (Mayan) in which certain numerals require classifiers whereas others appear only bare (Bale and Coon 2014). Furthermore, even in well-studied obligatory classifier languages such as Japanese certain quantifiers do not take classifiers (Sudo 2016). If the occurrence of the classifier was due to the semantics of nouns, those facts would be unaccounted for since there is no evidence that nouns change their meaning depending on a quantifier that modifies them. Consequently, according to the alternative explanation of the role of classifiers they are required to compensate semantic ‘deficits’ of numerals rather than nouns. The idea is that in classifier languages numerals lack a semantic component that would allow them to quantify over entities, recall (13-b) in Section 2.2. This role is, thus, assigned to classifiers.

This semantic view is also confirmed by a comparative study of the ordering of classifiers, nouns and numerals by Cinque (2020). What Cinque observes (building on previous work) is that classifiers never occur on the opposite side of the noun than the numeral; orders such as *NUMERAL > N > CLF or *CLF > N > NUMERAL are not found. This receives a natural explanation under the hypothesis that [NUMERAL + CLF] form a unit/constituent, which is what we are proposing here as well. Furthermore, as long as the classifier is adjacent to the numeral, it may in fact be separated from the noun by the numeral, yielding the order CLF > NUMERAL > N. Such examples are attested, see, e.g., the data from Upper Neacaxa Totonac in (36), or Hawaiian in (37). Both of these facts suggest that the syntactic structure contains a constituent [CLF NUMERAL], which does not split and it is (as a unit) positioned with respect to the noun.

With the crucial assumptions introduced, let us now move to the details of the proposal.

6.3 Semantic features

In order to capture the morphological complexity of basic cardinals discussed in Section 3 and 4 and summarized in Table 8 in Section 5, we propose three semantic primitives: SCALE, NUM (for ‘number’) and CL (for ‘classifier’), which we take to be cross-linguistically stable ingredients of the meaning of numerals.

The first feature, namely SCALE, denotes a closed interval, i.e., a set of natural numbers (type $\langle n, t \rangle$), see (54-a). As indicated by the subscript, the semantic content is lexically encoded, and thus specific for each distinct cardinal numeral. In other words, there is SCALE₁ for ‘one’, SCALE₂ for ‘two’ etc. What all SCALE elements have in common is that their lower bound is always 0. They differ in what the upper bound is. For instance, in the case of the cardinal *three*, SCALE₃ denotes the set of integers between 0 and 3, as in (54-b).

- (54) a. $\llbracket \text{SCALE}_m \rrbracket_{\langle n, t \rangle} = \lambda n_n [0 \leq n \leq m]$
 b. $\llbracket \text{SCALE}_3 \rrbracket = [0, 3]$

The second primitive is the syntactic head NUM. As such it is an invariant functional element shared by all cardinal numerals. The meaning of NUM is a function from intervals to numbers. In (55-a), the maximization operator MAX takes a set of integers denoted by SCALE and returns the greatest value from that set. As a result, it creates a name of a number concept (type n) compatible with abstract-counting contexts calling for numeric arguments. For instance, when applied to the interval $[0, 3]$ denoted by SCALE_3 , it will yield the number 3, see (55-b).

- (55) a. $\llbracket \text{NUM} \rrbracket_{\langle \langle n, t \rangle, n \rangle} = \lambda P_{\langle n, t \rangle} [\text{MAX}(P)]$
 b. $\llbracket \text{NUM} \rrbracket (\llbracket \text{SCALE}_3 \rrbracket) = 3$

The final component we propose is the CL head introduces object-counting semantics. Its role is to turn a number into a counting expression corresponding to that number. In particular, we postulate that the meaning of CL is a function from an integer to a predicate modifier supplied with the pluralization operation $*$ (Link 1983) and the measure function $\#(P)$ which maps a plurality of individuals to a numeric value corresponding to the number of individual making up that plurality, see (56-a) (Krifka 1989). In other words, CL is an expression of type $\langle n, \langle \langle e, t \rangle, \langle e, t \rangle \rangle \rangle$ that creates an object-counting numeral.^{23, 24}

- (56) a. $\llbracket \text{CL} \rrbracket_{\langle n, \langle \langle e, t \rangle, \langle e, t \rangle \rangle \rangle} = \lambda n_n \lambda P_{\langle e, t \rangle} \lambda x_e [*P(x) \wedge \#(P)(x) = n]$
 b. $\llbracket \text{CL} \rrbracket (\llbracket \text{NUM} \rrbracket (\llbracket \text{SCALE}_3 \rrbracket)) = \lambda P_{\langle e, t \rangle} \lambda x_e [*P(x) \wedge \#(P)(x) = 3]$

Combining the components in (54) and (55) in a compositional manner gives us the abstract-counting structure in (57). As one can see in the tree, the entire NUMP refers to a number concept. Such a denotation is thus compatible with expressions calling for numerical arguments such as *be a prime number* or *plus* which we take to be of type $\langle n, t \rangle$ and $\langle n, n \rangle$, respectively.

²³Notice, however, that nothing crucial hinges on the exact denotation of CL. What is essential is the fact that it shifts an object of type n into a quantifying expression. As long as this is ensured, the proposed system is also compatible with other theories of the object-counting meaning.

²⁴An anonymous reviewer points out that our denotation for the CL head (and its semantic type $\langle n, \langle \langle e, t \rangle, \langle e, t \rangle \rangle \rangle$) raises the question whether our system allows for the existence of classifiers that can be used without an accompanying numeral (e.g., *grain*). The general response is that care must be taken not to confuse the abstract meaning primitives like CL that we are describing in this section with particular morphemes (like *grain*). The reason is that we adopt here a realization approach to morphology (to be described shortly) where phonological strings like *grain* may first of all realize multiple meaning elements, and second, they may also represent different sets of such features in different uses. Therefore, there is no entailment that classifiers – when they occur without numerals – have exactly the same semantic type as when they occur with the numeral. This is because of the fact that at the core of any late-insertion approach lies the idea that individual phonological strings (like *grain*) can be inserted in multiple different environments. As a result, phonological strings (*grain*) may have no consistent semantic type across their different uses. For instance, the numeral *five* has a different type when it occurs as an abstract-counting numeral and when it occurs as an object-counting numeral.

$$\begin{array}{c}
(57) \quad \text{NUMP}_n \\
\quad \quad \quad 3 \\
\quad \quad \quad \swarrow \quad \searrow \\
\text{NUM}_{\langle\langle n,t \rangle, n \rangle} \quad \text{SCALE}_{3\langle n,t \rangle} \\
\lambda P_{\langle n,t \rangle}[\text{MAX}(P)] \quad \lambda n_n[0 \leq n \leq 3]
\end{array}$$

Adding the CL head as defined in (56) on top of (57) leads in turn to (58) which represents structure of the object-counting numeral. The semantic contribution of CL derives us a predicate modifier which when combined with a noun returns a set of pluralities in the denotation of that noun whose cardinality is the number referred to by the NUMP, e.g., 3 in the case of (58).

$$\begin{array}{c}
(58) \quad \text{CLP}_{\langle\langle e,t \rangle, \langle e,t \rangle \rangle} \\
\quad \quad \quad \lambda P_{\langle e,t \rangle} \lambda x_e[*P(x) \wedge \#(P)(x) = 3] \\
\quad \quad \quad \swarrow \quad \searrow \\
\begin{array}{c}
\text{CL}_{\langle n, \langle\langle e,t \rangle, \langle e,t \rangle \rangle \rangle} \\
\lambda n_n \lambda P_{\langle e,t \rangle} \lambda x_e[*P(x) \wedge \#(P)(x) = n]
\end{array}
\quad
\begin{array}{c}
\text{NUMP}_n \\
3 \\
\swarrow \quad \searrow \\
\text{NUM}_{\langle\langle n,t \rangle, n \rangle} \quad \text{SCALE}_{3\langle n,t \rangle} \\
\lambda P_{\langle n,t \rangle}[\text{MAX}(P)] \quad \lambda n_n[0 \leq n \leq 3]
\end{array}
\end{array}$$

Our model is broadly based on the idea that the meaning components postulated above are uniformly structured across languages and they must all be pronounced, though languages differ in how they pronounce them. So, for instance, the English phrase *three apples* and the Mandarin classifier construction *sān gè qiú* ‘three balls’ get the corresponding denotations in (59-a) and (59-b), respectively (cf. Krifka 1995).

$$\begin{array}{ll}
(59) & \text{a. } \llbracket \text{three apples} \rrbracket = \lambda x[*\text{APPLE}(x) \wedge \#(\text{APPLE})(x) = 3] \\
& \text{b. } \llbracket \text{sān gè qiú} \rrbracket = \lambda x[*\text{BALL}(x) \wedge \#(\text{BALL})(x) = 3]
\end{array}$$

In the next section, we will show that it is possible to use a cross-linguistically uniform structures given in (57) and (58) and still account for language variation in the domain at hand, summarized in Section 5.

7 Spellout

The main idea of our analysis is that the semantic structures proposed in the preceding section are universal, and variation among languages reduces to how these ingredients are pronounced in the individual languages. This idea is depicted in Table 9. In the table, we have six rows, one for each pattern. The table is further divided into two columns labeled ABSTRACT and OBJECT, which designate the relevant types of numerals. The abstract counting column has two components of meaning, each in its own sub-column. The object-counting column has three ingredients.

	ABSTRACT			OBJECT		
	SCALE	NUM		SCALE	NUM	CL
SIMPLEX SYNCRETIC	three		English 3	three		
SIMPLEX SUPPLETIVE	tnejn		Maltese 2	zewġ		
SIMPLEX STACKING	sān		Mandarin 3	sān	gè	
COMPLEX SYNCRETIC	xu	jal	Sanzhi Dargwa 5	xu	jal	
COMPLEX SUPPLETIVE	pah	w	Mokilese 4	pah	men	
COMPLEX STACKING	ru(ō)	vō	Vera’a 2	ru(ō)	vō	ne

Table 9: Analysis Overview

The basic idea is that all of these ingredients must be lexicalized, i.e., linked to a specific morpheme (or more generally, to a lexical item). Individual numeral roots in particular languages are then analyzed as lexicalizing a variable number of these components. When the numeral is able to lexicalize both the SCALE and NUM heads, we see just a single marker in the abstract counting column. These are the SIMPLEX patterns above the double horizontal line. When the numeral lexicalizes only the SCALE head, we get the COMPLEX numerals, because we need an additional marker to lexicalize NUM.

The SIMPLEX and COMPLEX pattern further separate into three different types depending on how the extra CL component is realized in the object-counting column. If we need an extra morpheme for it, we get the stacking pattern. If it is realized along with other meaning components, we get either the syncretic pattern or the suppletive pattern depending on details that we shall look more into as the discussion unfolds. At this point, the main message is that we can lexically specify individual morphemes for the ingredients they spell out, and by doing so, derive the morphological variation from this assumption. Needless to say, the lexical specification of morphemes is the one place in the grammar where evidence for cross-linguistic variation is indisputable.

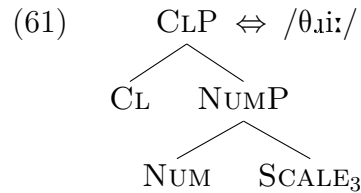
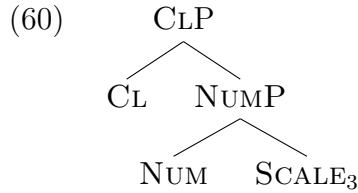
In the remainder of this section, we formalize this intuition using an independently established spellout algorithm developed within the Nanosyntax framework (Starke 2018). The very same algorithm has been used in the study of degree morphology (De Clercq and Vanden Wyngaerd 2017; Vanden Wyngaerd et al. 2020), relative/wh- pronouns (Wiland 2018, Bergsma 2019) and case morphology (Caha 2019). The relevant point is that the account does not represent an *ad hoc* approach. Rather, we use an independently established model of cross-linguistic variation in order to show that the morphological facts discussed up to now can be derived while assuming the invariant structure (58).

7.1 Simplex syncretic numerals

Nanosyntax is a realizational post-syntactic model of morphology. This means that syntactic structures are built first from abstract meaningful components, which are then mapped onto their pronunciation during the so-called lexicalization procedure. A crucial element in the lexicalization process is the lexicon, language-particular list of lexical items. In Nanosyntax, each lexical item is understood as a link between representations belonging to different modules. Concretely, they link a particular syntactic structure to a particular phonology and/or concept.

Due to the linking, lexical items serve as a ‘translation’ instructions of sorts. We can read them as follows: if syntax generates the stored representation S, then this representation is linked to a phonological representation P, which is associated to the lexical entry. Under this view, lexical entries have the restricted format $\langle S, P \rangle$, where each of S and P is a well-formed representation in the respective module.²⁵

In order to have a concrete lexical item as an example, consider the English numeral *three*. This numeral can act as an object-counting numeral. The structure of object-counting numerals according to our proposal is as given in (60). In order to express the fact that this structure can be pronounced by the numeral *three*, we link the relevant structure to the phonology $/\theta.i:/$. This linking is shown in (61). This numbered item corresponds to the lexical item for the numeral *three*.



As already said, the entry for *three* can be read as a ‘translation’ instruction: when syntax builds CLP, this structure can be spelled out as $/\theta.i:/$.

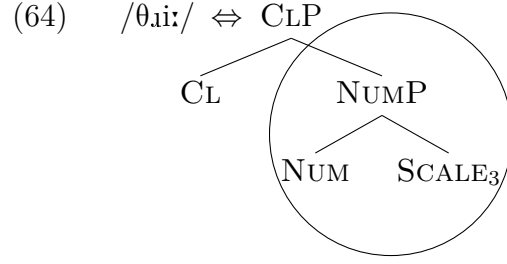
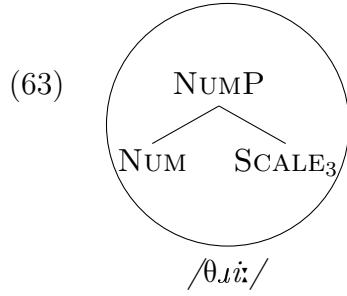
In Nanosyntax, the decision whether a lexical item can (or cannot) spell out a particular structure is dependent on the notion of matching. The rule is that whenever matching between a lexical entry and the structure obtains, the lexical entry can spell out the structure. Matching is based on identity: whenever the syntactic structure is identical to something stored in the lexicon, the item that contains the relevant tree matches the syntax tree. This is encoded by the so-called Superset Principle, see (62).

(62) THE SUPERSET PRINCIPLE (Starke 2009):

A lexically stored tree L matches a syntactic node S iff L contains the syntactic tree dominated by S as a subtree.

The Superset Principle leads to the consequence that a lexical entry matches any structure that is contained inside the lexical entry. This means that when syntax only builds the abstract-counting numeral – with structure as in (63) – the lexical item for the numeral *three* can be inserted, because it matches the structure. Matching (recall) is defined by The Superset Principle: since the lexical entry for the numeral (repeated in (64)) contains the structure (63), matching obtains and spellout can take place. Spellout is encoded by the circle around NUMP in (63). The circle in (64) just highlights the fact that the lexical entry literally contains a part that is identical to the structure spelled out.

²⁵Lexical items may of course be also associated to some concept C, but we disregard this here. The semantic information relevant for our purposes is not associated to lexical entries in some arbitrary manner, because it is an inseparable part the grammatical features the lexical entries spell out.

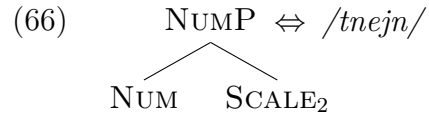
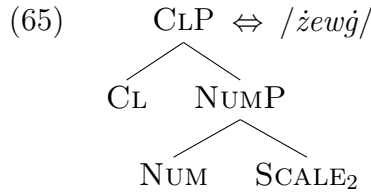


As a result, when a language has a numeral with an entry such as (64), we get a syncretism between the object- and abstract-counting function.

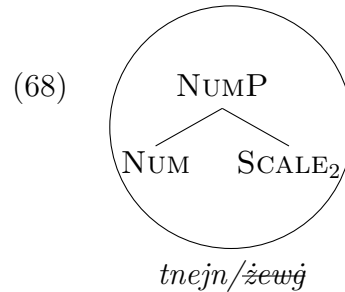
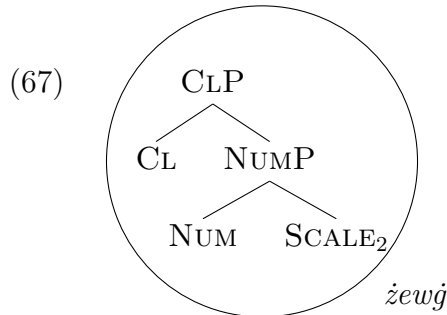
7.2 Simplex suppletive pattern

Let us now address the question how the simplex suppletive pattern arises. This pattern is found in languages where a single numeral (e.g., ‘two’) has two lexical entries, each for a different function. As an example, recall the Maltese numeral ‘2,’ which has the shape *tnejn* for abstract counting and *żewġ* for object counting (recall (28)-(29)).

The lexical entries are given below. The entry for the object counting numeral is shown in (65), the abstract-counting numeral is in (66).



With these entries in place, let’s consider the spellout of the object- and abstract-counting forms. These are depicted in (67) and (68) respectively. In the case of the object counting numeral, we only have one matching lexical item, namely *żewġ*. The reason is that the other numeral (*tnejn*) does not match the relevant constituent (it does not contain it). Therefore, the object-counting numeral is spelled out as *żewġ*.



Let us now turn to the abstract-counting structure (68). This structure is contained in both lexical entries. Therefore, both entries match, and they can both be used to spell out the structure. In such situation, competition among entries arises. The competition is resolved according to the so-called Elsewhere Condition (Kiparsky 1973), which demands that ‘the

more specific’ entry wins. In a system where insertion is governed by the Superset Principle, the most specific entry is the one that has fewer superfluous features. In the case of the abstract-counting structure depicted in (68), this is *tnejn*, whose entry has no superfluous features. On the other hand, *zewǵ* does have one superfluous feature and it therefore loses in the competition. In sum, the pair of lexical entries in (65)-(66) gives rise to the simplex suppletive pattern.

- (69) THE ELSEWHERE CONDITION: When multiple items match, chose the more specific one (the one that has fewer superfluous features).

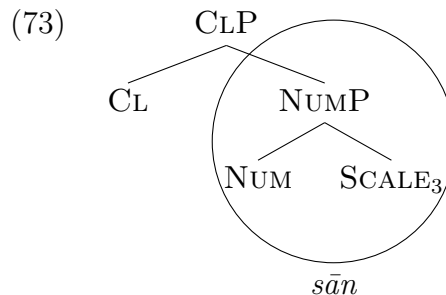
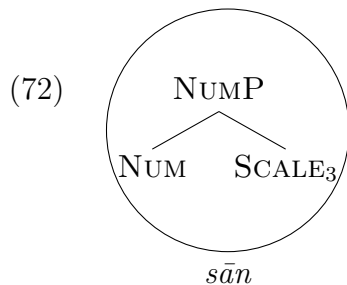
7.3 The simplex stacking pattern

In the simplex stacking pattern (which is found, for instance, in Mandarin), we encounter (for the first time) a situation where two morphemes are needed to spell out a particular structure. We shall begin by providing the lexical entries for the numeral ‘three’ (*sān*) and for the default classifier *gè*; see (70) and (71) respectively.

(70) $\text{NUMP} \Leftrightarrow /s\bar{a}n/$

(71) $\text{CLP} \Leftrightarrow /g\grave{e}/$

We can see that the lexical entry for the numeral in (70) corresponds to the abstract-counting structure. As a result, when syntax builds such a structure, as in (72), the numeral can be used to spell it out. As a result, we correctly model the fact that we only see the bare numeral in the abstract-counting function in Mandarin.



Consider now the object-counting structure in (73). Here, the numeral cannot spell out the full structure because it does not contain it in its entry. (Hence, no match and no spellout.)

The maximal constituent matched by the numeral is NUMP, circled in (73). This still leaves the head CL and its projection CLP without spellout. Leaving projections without spellout is not allowed in Nanosyntax: in order to externalize meaning, all features must be spelled out (cf. Fábregas 2007). The intuition which we shall follow is that the two unlexicalized nodes in (73) are spelled out by the classifier *gè*. These are also precisely the two nodes that are contained inside its entry in (71).

In order to see how this intuition is technically implemented in Nanosyntax, we must introduce the idea of cyclic spellout and spellout-driven movement. Cyclic spellout in Nanosyntax means that every time a new feature is introduced in the syntax, it must be lexicalized before the derivation is allowed to continue. We state this in (74).

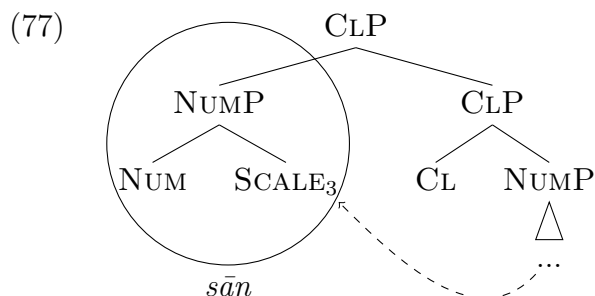
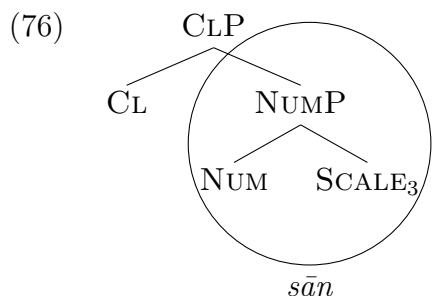
- (74) *Cyclic phrasal spellout.* Spell out must successfully apply to the output of every Merge F operation. Since Merge proceeds bottom-up, so does spellout. After successful spellout, the derivation may terminate, or proceed to another round of Merge F.

The consequence of adopting cyclic spellout is that structures are built in small steps, where each Merge F operation is immediately followed by spellout. For example, in order to build the structure of the object-counting numeral (corresponding to CLP), we must first build – and spell out – the abstract-counting numeral. Therefore, the tree in (72) can be considered the first step in the construction of the object-counting numeral in (73). The tree (73) (with a circle around NUMP) then represents a stage in the derivation where we have added a new feature to the structure (72), and we now need to spell out the new structure.

In order to see how exactly spellout operates, let us present in (75) the so-called spellout algorithm. This algorithm implements both the idea of Cyclic Spellout, and it also says how exactly spellout operates.

- (75) The Spellout Algorithm (Starke 2018)
- Merge F and spell out.
 - If (a) fails, move the Spec of the complement of F and spell out.
 - If (b) fails, move the complement of F and spell out.

The algorithm has several ranked options for spelling out the FP created by merging F. The first option is to find a match for that FP and spell it out. By hypothesis, the Mandarin numeral ‘two’ is stored as a NUMP, and it therefore does not match the whole tree in (76). What this means is that other derivational options in the algorithm (75) must be explored.



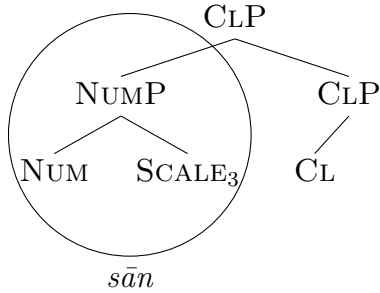
The first rescue option that must be tried (since spellout failed in (76)) is to move the Spec of the complement (and then retry spelling out F). However, in (76), the complement has no Spec, the option (75-b) is therefore undefined. As a consequence, the final derivational option is tried, see (75-c). This derivational option requires us to move the complement of F, which is a step we show in (77).

The root node in (77) has the same label as the one on the right branch. The point here is not so much to suggest that this is an adjunction structure; the point is to say that it is the right branch that provides the label to the root node.

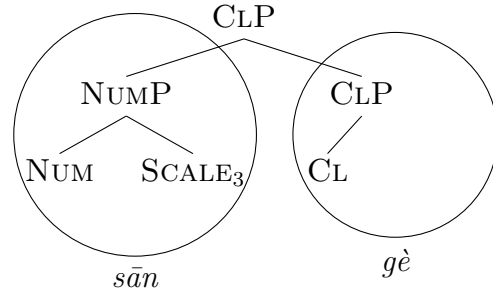
After complement movement, the structure (77) can be simplified as in (78) with the trace of the movement ignored. The assumption that spellout movements do not leave a trace is not crucial, even though it is a standard part of the Nanosyntax toolbox. The reason why

we follow this idea here is that ignoring the trace makes it easy to see when lexical entries match and when they don't. See Starke (2018) and Caha (2019) for more discussion.

(78)



(79)



The spellout algorithm now requires that after we move the complement, we try to spell out FP again. This time, matching is successful, because the general classifier *gè*, recall (71), can be inserted at the lower CLP, as in (79). As a result, the simplex stacking pattern is derived using the lexical entries in (70)-(71), and we also derive the correct ordering. The structure in (79) also replicates the intuition depicted in Table 9 to the effect that the classifier in Mandarin spells out CL, while the numeral root spells out SCALE and NUM.

7.4 On the interaction between suppletion and stacking in colloquial Mandarin

At this point, we have shown how all the three simplex patterns arise. The current section turns to a particularly interesting set of facts surrounding the numeral ‘three’ in spoken Mandarin, relying on the description in He (2015, 199) and Qi and He (2019, 76-8). We argue that the facts provide a particularly compelling case for analyzing the patterns at hand by means of phrasal spellout.

As a background, recall that Mandarin generally exhibits the stacking pattern illustrated in (80-a,b).

- (80) a. jiǔ chùyǐ sān shì sān.
 nine divide.by three COP three
 ‘Nine divided by three is three.’

- b. sān gè xuéshēng
 three CLF student
 ‘three students’

Mandarin

- c. sā (*gè) xuéshēng
 three.CLF CLF student
 ‘three students’

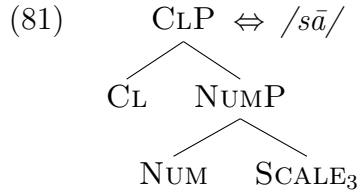
Colloquial Mandarin

Interestingly, the spoken language has developed a special form for the object-counting numeral ‘three,’ namely *sā*. This form is in (80-c), and He (2015, 199) describes it as a “reduced form” corresponding grammatically to the Standard Mandarin combination *sān gè* ‘three CLF.’ The analysis of *sā* as an opaque combination of a numeral and a classifier is motivated by the fact that *sā* must combine with the noun directly, i.e., without a classifier; see (80-c).

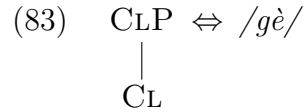
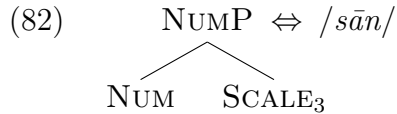
In our system, $s\bar{a}$ is therefore an instance of a suppletive object-counting numeral with $s\bar{a}n$ in (80-a) as its abstract-counting counterpart.

The important point is that this description nicely illustrates the gist of our idea as to how morphologically simplex object-counting numerals arise, namely as an opaque realization of a complex structure containing the meaning components corresponding to the combination of an abstract-counting numeral and a classifier.

We will see shortly that the reduction of $s\bar{a}n$ $g\bar{e}$ ‘three CLF’ to $s\bar{a}$ is not phonological, but morphological in nature. In our proposal, the colloquial example in (80-c) therefore requires us to posit a special lexical entry for $s\bar{a}$, which only exists in the colloquial variety. The entry is as given in (81).



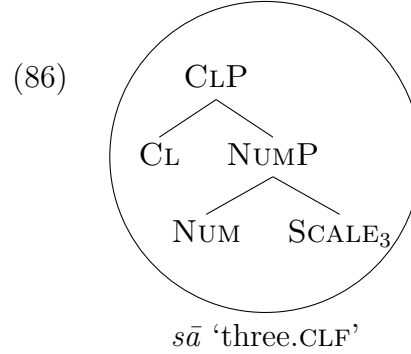
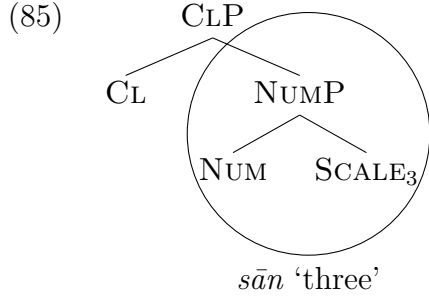
This lexical entry co-exists in the colloquial variety with both the lexical entry for the abstract-counting $s\bar{a}n$ ‘three,’ which the colloquial variety uses in arithmetical statements just like the standard variety. It also coexists with the lexical entry for the general classifier $g\bar{e}$, which is still used in the colloquial variety with other numerals. In sum, the lexical entry (81) is found in the lexicon of colloquial Mandarin alongside the lexical entries given in (82) and (83), repeated from (70) and (71) respectively.



The coexistence of the three lexical entries in the colloquial variety allows us to illustrate the explanatory power of the theoretical tools that we have posited up to now. First of all, note that $s\bar{a}$ with the lexical entry as in (81) could be in principle used in mathematical statements, because it contains the abstract-counting structure. However, $s\bar{a}n$ with the entry (82) is a perfect match for the abstract-counting structure, and we therefore predict that $s\bar{a}$ cannot occur in arithmetical environments. This prediction is borne out, see (84) (Qi and He 2019, 77).

- (84) *Wǒ zuì xǐhuan de shù shì $s\bar{a}$.
 1.SG most like PRT number COP three.CLF
 Intended: ‘The number I like most is three.’ Mandarin

A more interesting prediction relates to the relative sequence of steps in the spellout algorithm. To see it, suppose that the cyclic derivation has produced the abstract-counting numeral $s\bar{a}n$ ‘three.’ Suppose now that the head CL is added, as in (85), repeated from (76).

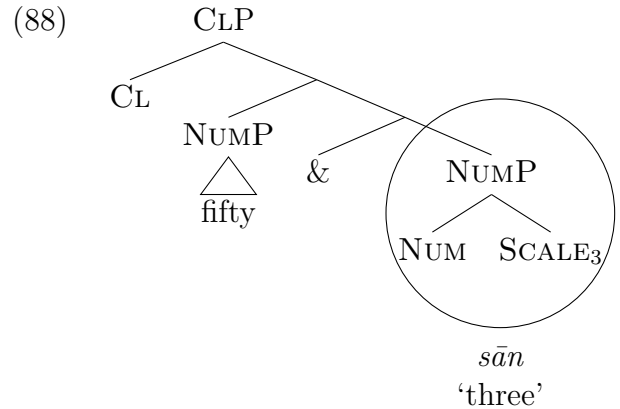
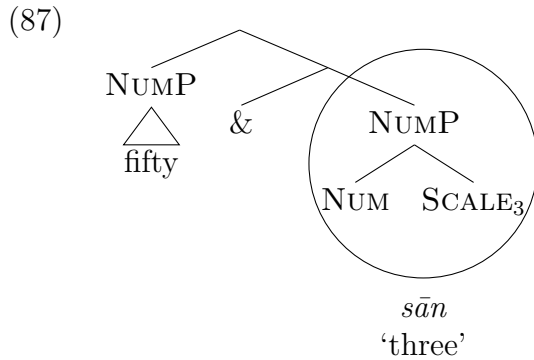


The prediction – related to the ordering of steps in the spellout algorithm – is that when confronted with (85), the system first tries to spell out the whole constituent without movement. In the colloquial variety, this attempt leads to a successful match by *sā*, see (86). As a result, we predict that in colloquial Mandarin, *sā* (rather than *sān gè*) is going to be the spellout of (85), which is the case.

On the other hand, if complement movement was ordered before direct spellout, we would derive the form with the classifier (i.e., *sān gè*), despite the presence of the suppletive portmanteau *sā* ‘three.CLF’. Therefore, if the ordering of the steps were different, we would fail to derive the colloquial form even if it was contained in the lexicon, clearly the wrong result. In sum, we found support for the proposed ordering of the two relevant steps of the spellout algorithm.

Most curiously, some additional facts show that even though colloquial Mandarin prefers the mono-morphemic spellout *sā* ‘three.CLF’ over *sān gè* ‘three CLF,’ this is only so if *sān* and the CL head form a constituent. Since this is something that is predicted by the phrasal-spellout account of suppletion, let us discuss the relevant facts in the remainder of this section.

The relevant configuration arises when the numeral ‘three’ is a part of a transparent compound numeral, such as the Mandarin equivalent of ‘fifty **three**.’ The structure of such numerals argued for in He (2015, 200) is as in (87). In this tree, we only elaborate on the fine structure of *sān* ‘three,’ while the rest of the structure is simplified. The head that contributes the additive semantics is represented as &, following He.



Against this background, consider what our theory predicts when the CL head is merged on

top of the compound-numeral structure, as in (88). The spellout algorithm requires that CL must be spelled out. The first attempt is to spell out CLP directly. This will fail in (88): even though the numeral *sā* can spell out CL along with the NUMP, the matching principle allows this only if the CL head and NUMP are contained in the lexical entry of the numeral; recall the Superset Principle (62). This is obviously not the case in (88); therefore, we predict that *sā* ‘three.CLF’ cannot appear in the compound numeral ‘fifty three.’

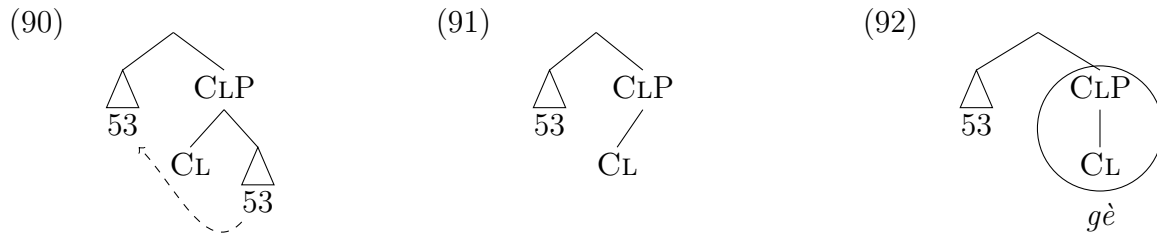
The prediction is borne out. As the data in (89-a) show, it is indeed impossible to use *sā* ‘three.CLF’ inside the compound numeral. Instead, the analytic combination *sān gè* is found, as in (89-b).

- (89) a. *wǔ shí **sā** xuéshēng
 five ten three.CLF student
 ‘fifty three students’
 b. [wǔ shí **sān**] **gè** xuéshēng
 five ten three CLF student
 ‘fifty three students’

Colloquial Mandarin

The example (89-b) (where *sān* is directly followed by *gè*) shows that the numeral *sā* does not arise as a result of a simple phonological or morphological rule that operates on the basis of linear adjacency between *sān* ‘three’ and *gè* ‘CLF.’ Rather, it operates on the basis of syntactic constituency, which is predicted by our account where suppletive object-counting numerals like *sā* arise as a result of phrasal spellout.

Finally, let us turn to the issue of how we can generate the correct form in (89-b). Note that when the spellout algorithm fails to directly spell out CL in (89), it proceeds to movements. The first movement to be tried is Spec-movement, recall (75-b). The phrase in (88) has a Spec, namely the phrase ‘fifty.’ However, even if that phrase is removed, the & head (responsible for addition) is in the way of successful spellout of CL and the NUMP. Therefore, the option that ultimately succeeds is the one that moves the complement of the CL head to the left, as schematically depicted in (90). This gives rise to the structure in (91) (with the trace removed), and the remnant CLP is spelled out by the classifier, see (92).

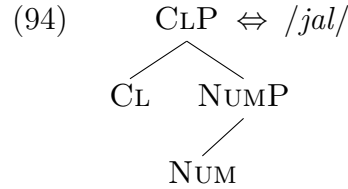


As a final point in this section, we note (following He 2015) that the data discussed above support the idea presented in Section 3.4 that in at least some cases, compound numerals form a complex constituent composed of abstract counting numerals. Only after the compound numeral is formed, CL is merged in order to turn the abstract-counting compound numeral into the corresponding object-counting numeral. (This does not exclude the possibility that in other cases, the structure may be more complex, recall footnote 12.)

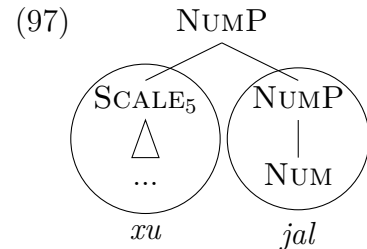
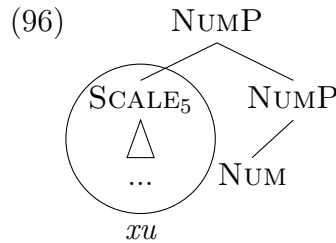
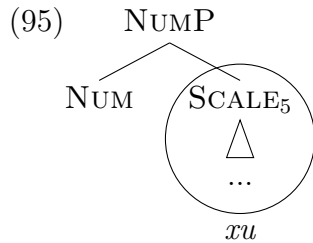
7.5 Complex syncretic numerals

In this section, we turn to COMPLEX numerals. Recall that complex numerals are bi-morphemic already at the level of abstract-counting numerals. In the current model, this entails that they spell out separately SCALE and NUM. Consider, for instance, the case of the Sanzhi Dargwa numeral *xu-jal* ‘five.’ This complex numeral can be used both as an abstract-counting numeral and also as an object-counting numeral, recall (39)-(40). We have therefore classified it as a COMPLEX SYNCRETIC numeral.

The lexical entries needed to capture this pattern are given in (93)-(94). The first lexical entry (namely *xu* ‘five’) in (93) spells out just SCALE (reflecting the intuition in Table 9). The remaining features are included in the lexical entry of *jal*, see (94).²⁶



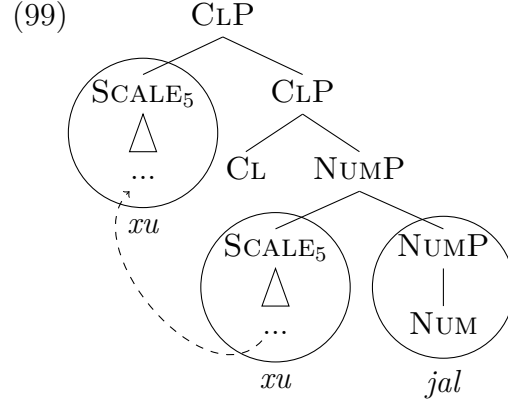
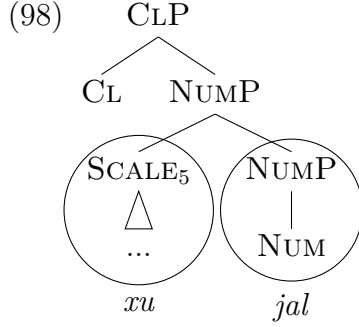
In order to see how these lexical entries lead to the result *xu-jal* in both object- and abstract-counting uses, consider the derivation below. We start by assembling the lowest SCALE₅ node and we spell it out as *xu*, using the lexical item (93). Once the spellout of SCALE₅ succeeds, the abstract-counting structure is derived by the addition of NUM, see (95).



The NUMP in (95) fails to spell out, because there is no match. (Neither the entry for *xu* (93) or *jal* (94) contain the structure.) Therefore, we must use the other options of the spellout algorithm, specifically complement movement. (96) shows the configuration after SCALE₅ undergoes complement movement from below NUM (the trace is ignored). In this configuration, the remnant NUMP node may be spelled out by *jal*. This correctly yields the form *xu-jal* in (97) as the spell out of the abstract counting structure.

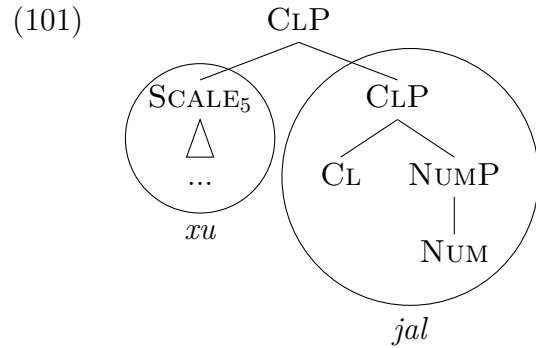
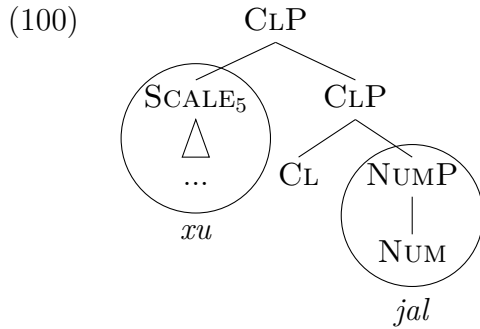
Consider now how the object counting numeral is derived. Following the algorithm, we first merge the feature CL on top of the abstract-counting numeral. The abstract counting numeral is in (97), and merging CL yields (98). Spellout without movement fails here, since the top node of (98) (CLP) is not contained inside any single entry.

²⁶In order for the spellout procedure to work smoothly, we are led to assume in (93) that the SCALE node is actually complex and arises by merging two ingredients together. This complexity is reflected by the triangle under SCALE in (93). The reason why we have to assume this is theory internal. It has to do with the way the spellout algorithm works: it always starts by merging two features together, and only then it pronounces the resulting FP. Therefore, if a morpheme pronounces SCALE, the node must be internally complex. We do have some ideas about what these ingredients could be, but we shall not elaborate on this idea in any detail; see Vanden Wyngaerd et al. (2020) for a decomposition of adjectival scales that may be relevant here.



Therefore, as customary by now, other clauses of the spellout algorithm are invoked. The first option to be tried is Specifier movement, recall (75-b). The movement is shown in (99).

The result of Spec movement (with the trace ignored) is given in (100). After Spec movement, we have to retry ‘spell out FP,’ where FP corresponds to the remnant CLP. This CLP is contained inside the lexical entry of the suffix *jal*, and the suffix is therefore inserted, see (101). Note that as a part of the process, the ‘old’ spellout of NUMP in (100) is replaced by the ‘new’ spellout of CLP in (101). This is called overriding in Nanosyntax; see Starke (2009) and other Nanosyntax references for more on overriding, here we only make this explicit.



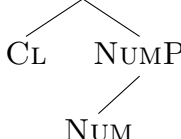


Before we conclude, let us make a general remark about syncretism. In the current theory, syncretism is always the result of the fact that one and the same lexical item may spell out two different structures. In the simplex syncretic pattern (e.g., *three*), the numeral would either spell out all three meaning components of object-counting structures, or just NUM and SCALE in the abstract-counting use. In the complex pattern, this is very similar. The ambiguity boils down to the fact that the suffix *-jal* either spells out only the NUM feature, or NUM+CL. This is also as coded in Table 9.

7.6 Complex suppletive numerals

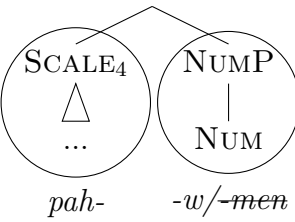
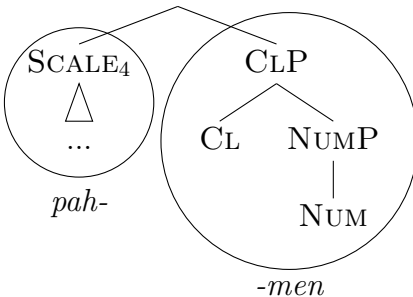
Let us now consider COMPLEX SUPPLETIVE numerals. The derivation of this type is very similar to the way COMPLEX SYNCRETIC numerals work, the only difference is that instead of having an ambiguous suffix (like *jal*), we will have two different suffixes: one for NUM+CL, and the other one for NUM only.

The lexical entries needed for this are given below. They are modeled on the basis of Mokilese, recall (48)-(50). In Mokilese, we find the general classifier *-w* in abstract-counting contexts, and we therefore hypothesize that numerals spell out only SCALE_4 , see (102). The general classifier *-w* spells out NUM, see (103). In (104), we give the lexical entry for the classifier *men*, which is suffixed to the numeral in an object-counting function when animates are counted. The crucial point is that *-men* replaces *-w*, and we therefore attribute to it both the feature CL and NUM.

- (102) $\text{SCALE}_4 \Leftrightarrow /pah/$

 ...
- (103) $\text{NUMP} \Leftrightarrow /-w/$

 NUM
- (104) $\text{CLP} \Leftrightarrow /-men/$

 CL NUMP
 NUM

The entries as given above will derive a pure suppletive system – which (as the reader may recall from the discussion in Section 4.3) we have not been able to find yet. For the sake of the discussion, we will nevertheless show how the derivation of such a pattern proceeds, using the entries as given above. Later, we shall also derive the actual attested patterns that mix suppletion and syncretism. (That will require us to say something about the difference between different classifiers.)

The derivation of a pure suppletive type proceeds in exactly the same steps as in the preceding section 7.5. First, SCALE is spelled out by the numeral. The feature NUM is added. There is no single morpheme to spell out both, therefore SCALE moves to the left and the remnant NUMP is spelled out by *-w*, see (105). Note that *-men* can also be inserted here, but it loses in competition with *-w*. Note also that the structure has the same abstract shape as the one used for Sanzhi Dargwa, recall (97).

- (105) 
 SCALE_4 NUMP
 ... NUM
pah- *-w/-men*
- (106) 
 SCALE_4 CLP
 ... CL NUMP
 NUM
-men

The tree in (106) represents the object-counting numeral. It is abstractly also the same structure as in Sanzhi Dargwa, recall (101). The only difference between Mokilese and Sanzhi Dargwa is that Mokilese has a dedicated suffix for each of the right branches in (105) and (106), while Sanzhi Dargwa had just a single marker for both. This is then how an idealized complex suppletive pattern arises.

Now in reality, recall that the general classifier *-w* can also appear (as an ‘unmarked’ classifier) in object-counting use, recall (49-a). Therefore, we will have to attribute to it a lexical entry capable of spelling out CL, which makes it – as things stand – identical to

Sanzhi Dargwa *-jal*. This is empirically correct, since both morphemes can actually be used in both object- and abstract-counting uses.

However, by attributing CL to *-w*, we also make it indistinguishable from *-men*, which is wrong; *-men* does not appear in abstract-counting contexts. The answer we suggest here emanates from the observation that classifiers themselves may be morphologically complex. To see that, consider the following dataset from Nepali. Nepali is a simplex stacking language. In the abstract counting use, numerals are bare, see (107) (Turnbull 1982, 45).

- (107) dui ek-an **dui**
 two one-times two
 ‘Two times one is two.’ Nepali

When numerals are used in object-counting function, they must be accompanied by a classifier. (108-a) shows the general classifier *wota*, while (108-b) exemplifies the human classifier *jana*.

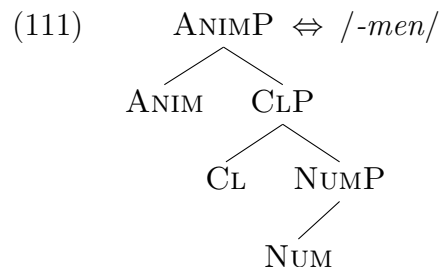
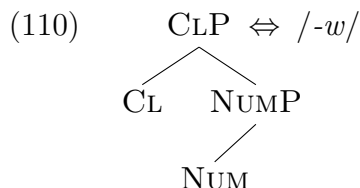
- (108) a. ek wota kaka
 one CLF.GENERAL uncle
 ‘one uncle’
 b. ek jana kaka
 one CLF.HUMAN uncle
 ‘one uncle’

Interestingly, as pointed out by Allasonnière-Tang and Kilarski (2020, 129-130), the general classifier (though not other classifiers) agrees in gender with the head noun, and shows up in two different shapes, namely *wot-a* for masculines and *wot-i* for feminines.

- (109) a. tin wot-a keto
 three CLF.GENERAL-M boy
 ‘three boys’
 b. tin wot-i keti
 three CLF.GENERAL-F girl
 ‘three girls’

What these data show is that the general classifier *wot-* (the realization of a pure CL node), may be followed by additional (agglutinative) markers that further restrict the applicability of the general classifier to masculine/feminine nouns respectively.

Extrapolating this observation to Mokilese, we may think of the non-general classifiers (that are opposed to a general classifier) as lexicalizing (in a manner of phrasal spellout) additional features apart from CL. For instance, the Mokilese situation could be modeled as a combination of the following two entries, see (110) and (111).



(110) shows the updated entry for the general classifier. Compared to the original version in (103), it can also spell out the CL head. With this lexical entry, it can appear in both object- and abstract-counting environments, cf. the Sanzhi Dargwa *jal* in (94). The animate classifier *men* has now been updated to include an additional ANIM feature, which (unlike the general classifier *-w*) imposes an additional restriction on the nature of the counted object. Note that since the classifier *-men* also includes the features NUM and CL, it does not stack on top of the general classifier, but replaces it.

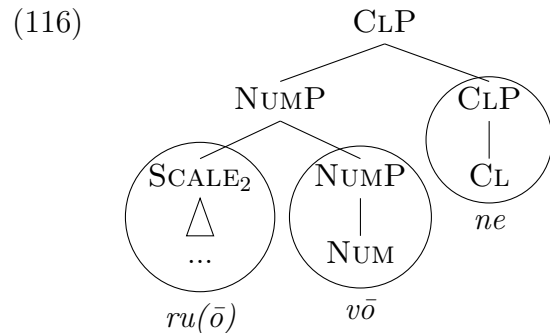
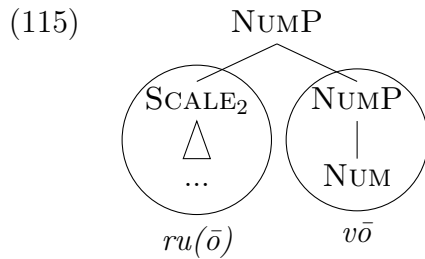
7.7 Complex stacking

The final type to be addressed is the complex stacking type. In this type of language, exemplified by Vera'a, the abstract numeral is complex (*vō-ru(ō)* 'PRFX-two'), and an additional marker stacks on top of the numeral in the object counting function (*ne vō-ru(ō)*). This can be modeled by proposing that each marker spells out exactly one feature as specified below.

$$\begin{array}{lll}
 (112) & \text{SCALE}_2 \Leftrightarrow /ru(\bar{o})/ & (113) \quad \text{Nump} \Leftrightarrow /v\bar{o}/ \\
 & \triangle & \text{NUM} \\
 & \dots & \\
 & & \text{CL} \\
 & & \text{CLP} \Leftrightarrow /ne/
 \end{array}$$

We shall now run the derivation using the markers as specified above. As a caveat, we point out that if we take the entries to be as in (112)-(114), we shall generate a mirror-image suffixal paradigm (i.e., instead of *ne vō-ru(ō)* we shall get *ne ru(ō)-vō ne*). This can be fixed by changing the shape of lexical entries into a different format that leads to prefixation (cf. Starke 2018, De Clercq and Vanden Wyngaerd 2018), but we do not take up this issue for reasons of space, since the derivation of prefixal paradigms in Nanosyntax would take us too far afield.

With this caveat in place, our starting point is the fact that the derivation of the abstract-counting numeral will proceed as in Mokilese and Sanzhi Dargwa, yielding the tree in (115). The reason for this is that the numeral can spell out only SCALE; for NUM to be spelled out, SCALE must move to the left, and the remnant Nump is spelled out by the marker *vō*.



Now the difference between Vera'a and the other languages is that CL is spelled out by a separate marker still, see (114). (In Mokilese and Sanzhi Dargwa, CL was spelled out jointly with NUM.) This leads to the fact that the derivation is forced to another step of complement movement, producing the completely analytic structure in (116).

8 Conclusions

In this article, we set out to ask the following question: what is the relationship between object-counting and abstract-counting numerals? Through the investigation of a number of languages, we concluded that the answer to this question is that object-counting numerals both syntactically and semantically contain abstract-counting numerals.

In order to model all the morphological effects that we encountered while comparing the two types of numerals, we have decomposed the numerals into three components, that we label SCALE, NUM and CL. According to our proposal, CL is missing in abstract-counting numerals, but it is found in object-counting numerals. This approach is in line with some earlier proposals such as Krifka (1995) and Hackl (2000), who derive the object-counting meaning of numerals from a number-denoting core.

We have further proposed that different patterns of marking come about as a result of the fact that numerals across different languages (and sometimes also within individual languages) lexicalize different sets of the three underlying formatives. We have provided an algorithmic implementation of this idea within the Nanosyntax framework.

Acknowledgments

We would like to thank the audiences at TripleA 7, SinFonIJA 13, CLS 57, AFLA 28, Olinco 5 as well as Bartosz Wiland and two anonymous reviewers for their helpful questions and comments. Special thanks go to our informants and consultants: Mary Chimaobi Amaechi, Albert J. Borg, Viacheslav Chirikba, Maia Duguine, Nina Haslinger, Chang Liu, Catriona Malau, Stefan Schnell, Yasu Sudo, and Tue Trinh. All errors are, of course, my own responsibility. Marcin Wągiel’s work was supported by a Czech Science Foundation (GAČR) grant to the Department of Linguistics and Baltic Languages at the Masaryk University in Brno (GA20-16107S).

References

- Aikhenvald, A. Y. (2000). *Classifiers: A Typology of Noun Categorization Devices*. Oxford: Oxford University Press.
- Ajiboye, O. (2005). *Topics on Yorùbá Nominal Expressions*. Ph. D. thesis, The University of British Columbia.
- Ajiboye, O. (2016). The Yorùbá numeral system. In O.-m. Ndimele and E. Chan (Eds.), *The Numeral Systems of Nigerian Languages*, pp. 1–25. Port Harcourt: M & J Grand Orbit.
- Allasonnière-Tang, M. and M. Kilarski (2020). Functions of gender and numeral classifiers in nepali. *Poznan Studies in Contemporary Linguistics* 56(1), 113–168.
- Bale, A. and J. Coon (2014). Classifiers are for numerals, not for nouns: Consequences for the mass/count distinction. *Linguistic Inquiry* 45(4), 695–707.
- Barz, R. K. and A. V. Diller (1985). Classifiers and standardization: Some South and South-East Asian comparisons. In D. Bradley (Ed.), *Papers in South-East Asian Linguistics*

- No. 9: *Language Policy, Language Planning and Sociolinguistics in South-East Asia*, pp. 155–184. Canberra: Australian National University.
- Beck, D. (2004). *A grammatical sketch of Upper Necaxa Totonac*. Munich: LINCOM.
- Bergsma, F. (2019). Mismatches in free relatives—grafting nanosyntactic trees. *Glossa: a journal of general linguistics* 4(1).
- Borer, H. (2005). *Structuring Sense I: In Name Only*. Oxford: Oxford University Press.
- Borg, A. (1974). Maltese numerals. *Zeitschrift der Deutschen Morgenländischen Gesellschaft* 124(2), 291–305.
- Borg, A. J. (1987). To be or not to be a copula in Maltese? *Journal of Maltese Linguistics* 17(18), 54–71.
- Bultinck, B. (2005). *Numerous Meanings: The Meaning of English Cardinals and the Legacy of Paul Grice*. Amsterdam: Elsevier.
- Caha, P. (2019). The Nanosyntax of case competition. lingbuzz/004875.
- Chierchia, G. (1998). Plurality of mass nouns and the notion of ‘semantic parameter’. In S. Rothstein (Ed.), *Events and Grammar*, pp. 53–103. Dordrecht: Kluwer.
- Chirikba, V. A. (2003). *Abkhaz*. Muenchen: LINCOM.
- Cinque, G. (2020). Word order variation: towards a restrictive theory. Manuscript, University of Venice.
- Corbett, G. G. (1978). Universals in the syntax of cardinal numerals. *Lingua* 46(4), 355–368.
- De Clercq, K. and G. Vanden Wyngaerd (2017). *ABA revisited: evidence from Czech and Latin degree morphology. *Glossa* 2(1), 69: 1–32.
- De Clercq, K. and G. Vanden Wyngaerd (2018). Unmerging analytic comparatives. *Jezikoslovlje* 19(3), 341–363.
- Elbert, S. H. and M. K. Pukui (1979). *Hawaiian grammar*. Honolulu: University of Hawaii.
- Fábregas, A. (2007). An exhaustive lexicalisation account of directional complements. In M. Bašić, M. Pantcheva, M. Son, and P. Svenonius (Eds.), *Nordlyd, special issue on Space, Motion, and Result*, Number 34.2 in Tromsø Working Papers on Language and Linguistics, pp. 165–199. Tromsø: CASTL, University of Tromsø.
- Forker, D. (2020). *A grammar of Sanzhi Dargwa*. Berlin: Language Science Press.
- Greenberg, J. H. (1978). Generalizations about numeral systems. In J. H. Greenberg (Ed.), *Universals of Human Language*, Volume 3, pp. 249–295. Stanford, CA: Stanford University Press.
- Hackl, M. (2000). *Comparative Quantifiers*. Ph. D. thesis, Massachusetts Institute of Technology.
- Harrison, S. (2019). *Mokilese reference grammar*. University of Hawaii Press.
- He, C. (2015). Complex numerals in Mandarin Chinese are constituents. *Lingua* 164, 189–214.

- Hewitt, G. (2010). *Abkhaz: A Comprehensive Self-Tutor*. Muenchen: LINCOM.
- Hundius, H. and U. Kölver (1983). Syntax and semantics of numeral classifiers in Thai. *Studies in Language* 7, 165–214.
- Hurford, J. R. (1998). The interaction between numerals and nouns. In F. Plank (Ed.), *Noun Phrase Structure in the Languages of Europe*, pp. 561–620. Berlin: Mouton de Gruyter.
- Hurford, J. R. (2001). Languages treat 1-4 specially. *Mind & Language* 16(1), 69–75.
- Ionin, T. and O. Matushansky (2006). The composition of complex cardinals. *Journal of Semantics* 23, 315–360.
- Ionin, T. and O. Matushansky (2018). *Cardinals: The Syntax and Semantics of Cardinal-Containing Expressions*. Cambridge, MA: MIT Press.
- Jacques, G. (2018). A grammar of Japhug. Manuscript.
- Kalita, J. (2011). The referring systems and the determinative elements of noun phrases in Assamese. In G. Hyslop, S. Morey, and M. W. Post (Eds.), *North East Indian Linguistics* 3, pp. 173–196. New Delhi: Cambridge University Press India.
- Kennedy, C. (2001). Polar opposition and the ontology of ‘degrees’. *Linguistics and Philosophy* 24(1), 33–70.
- Kennedy, C. (2015). A ‘de-Fregean’ semantics (and neo-Gricean pragmatics) for modified and unmodified numerals. *Semantics & Pragmatics* 8, 1–44.
- Kiparsky, P. (1973). ‘Elsewhere’ in phonology. In S. Anderson and P. Kiparsky (Eds.), *Festschrift for Morris Halle*, pp. 93–106. New York: Holt, Rinehart & Winston.
- Klamer, M. (2010). *A Grammar of Teiwa*. Berlin: Mouton de Gruyter.
- Klamer, M., A. Schapper, G. Corbett, G. Holton, F. Kratochvíl, and L. C. Robinson (2014). Numeral words and arithmetic operations in the Alor-Pantar languages. In M. Klamer (Ed.), *The Alor-Pantar Languages: History and Typology*, pp. 337–373. Berlin: Language Science Press.
- Krifka, M. (1989). Nominal reference, temporal constitution and quantification in event semantics. In R. Bartsch, J. van Benthem, and P. von Emde Boas (Eds.), *Semantics and Contextual Expression*, pp. 75–115. Dordrecht: Foris Publications.
- Krifka, M. (1995). Common nouns: A contrastive analysis of Chinese and English. In G. N. Carlson and F. J. Pelletier (Eds.), *The Generic Book*, pp. 398–411. Chicago: University of Chicago Press.
- Landman, F. (2003). Predicate-argument mismatches and the adjectival theory of indefinites. In *From NP to DP: Volume 1: The Syntax and Semantics of Noun Phrases*, pp. 211–237. Amsterdam: John Benjamins.
- Link, G. (1983). The logical analysis of plural and mass nouns: A lattice-theoretical approach. In R. Bäuerle, C. Schwarze, and A. von Stechow (Eds.), *Meaning, Use, and Interpretation of Language*, pp. 302–323. Berlin: Mouton de Gruyter.
- Malau, C. (2016). *A Grammar of Vurës, Vanuatu*. Boston: Walter de Gruyter.

- Nouwen, R. (2016). Making sense of the spatial metaphor for number in natural language. Unpublished manuscript. Available online on LingBuzz.
- Obikudo, E. F. (2016). Counting: The ịbani way. In O.-m. Ndimele and E. Chan (Eds.), *The Numeral Systems of Nigerian Languages*, pp. 217–223. Port Harcourt: M & J Grand Orbit.
- Pham, G. and K. Kohnert (2008). A corpus-based analysis of Vietnamese ‘classifiers’ con and cái. *Mon-Khmer Studies* 38, 161–171.
- Po-Ching, Y. and D. Rimmington (2015). *Chinese: A comprehensive grammar*. Routledge.
- Pukui, M. K. and S. H. Elbert (1986). *Hawaiian Dictionary: Hawaiian-English, English-Hawaiian*. Honolulu: University of Hawaii.
- Qi, J. and C. He (2019). The morphosyntax of numerals ɗi³³/ɗi³⁵ ‘one’ in Shuhi and implications for the semantics of numerals. *Lingua* 225, 63–80.
- Robinson, L. C. and J. W. Haan (2014). Adang. In A. Schapper (Ed.), *The Papuan Languages of Timor, Alor and Pantar. Volume 1: Sketch Grammars*, pp. 221–284. Berlin: Walter de Gruyter.
- Rothstein, S. (2010). Counting and the mass/count distinction. *Journal of Semantics* 27(3), 343–397.
- Rothstein, S. (2017). *Semantics for Counting and Measuring*. Cambridge: Cambridge University Press.
- Royer, J. (2017). Noun and numeral classifiers in Chuj (Mayan). In *Proceedings of the Berkeley Linguistics Society*, Volume 2, pp. 29–38.
- Schnell, S. (2011). *A Grammar of Vera’a, an Oceanic Language of North Vanuatu*. Kiel: Kiel University.
- Schwarzschild, R. and K. Wilkinson (2002). Quantifiers in comparatives: A semantics of degree based on intervals. *Natural Language Semantics* 10(1), 1–41.
- Scontras, G. (2013). Accounting for counting: A unified semantics for measure terms and classifiers. In T. Snider (Ed.), *Proceedings of Semantics and Linguistic Theory* 23, pp. 549–569. Ithaca, NY: CLC Publications.
- Seuren, P. A. (1984). The comparative revisited. *Journal of Semantics* 3(1), 109–141.
- Starke, M. (2009). Nanosyntax: A short primer to a new approach to language. *Nordlyd* 36(1), 1–6.
- Starke, M. (2018). Complex left branches, spellout, and prefixes. In L. Baunaz, K. De Clercq, L. Haegeman, and E. Lander (Eds.), *Exploring Nanosyntax*, pp. 239–249. Oxford: Oxford University Press.
- Storch, A. (2014). Counting chickens in Luwo. In A. Storch and G. J. Dimmendaal (Eds.), *Number-constructions and semantics. Case studies from Africa, Amazonia, India and Oceania*, pp. 265–282. Amsterdam: John Benjamins.
- Sudo, Y. (2016). The semantic role of classifiers in Japanese. *Baltic International Yearbook of Cognition, Logic and Communication* 11(1), 1–15.

- Turnbull, A. (1982). *Nepali Grammar & Vocabulary*. Asian Educational Services.
- Vanden Wyngaerd, G., M. Starke, K. De Clercq, and P. Caha (2020). How to be positive. *Glossa* 5(1), 23.
- Wiland, B. (2018). A note on lexicalizing ‘what’ and ‘who’ in Russian and in Polish. *Poznan Studies in Contemporary Linguistics* 54(4), 573–604.
- Wągiel, M. and P. Caha (2020). Universal semantic features and the typology of cardinal numerals. *Catalan Journal of Linguistics* 19, 199–229.
- Yue, A. O. (2017). The Sinitic languages: Grammar. In *The Sino-Tibetan Languages*, pp. 114–163. London: Routledge.