

An algebraic approach to translating Japanese

Valentin Boboc
University of Manchester

ABSTRACT

We use Lambek’s pregroups and the framework of compositional distributional models of language (“DisCoCat”) to study translations from Japanese to English as pairs of functors. Adding decorations to pregroups, we show how to handle word order changes between languages.

Keywords:
pregroups,
type grammars,
translations,
functorial

INTRODUCTION

1

Language has the purpose of conveying meaning. It is traditionally viewed as possessing both an empirical aspect – one learns language by practising language – and a compositional aspect – the view that the meaning of a complex phrase is fully determined by its structure and the meanings of its constituent parts.

In order to efficiently exploit the compositional nature of languages, a popular way of modelling natural languages is a categorical compositional distributional model, abbreviated “DisCoCat” (Coecke *et al.* 2010). Languages are modelled as functors from a category that interprets grammar (“compositional”) to a category that interprets semantics (“distributional”).

The compositional part is responsible for evaluating whether phrases or sentences are well-formed by calculating the overall grammatical type of a phrase from the grammatical types of its individual parts. There are several algebraic methods for modelling the

grammar of a natural language. In the present article we choose the well-established model of pregroup grammars. Pregroups were introduced in Lambek 1997 to replace the algebra of residuated monoids in order to model grammatical types, their juxtapositions, and reductions. Pregroup calculus has been applied to formally represent the syntax of several natural languages such as: French (Bargelli and Lambek 2001a), German (Lambek and Preller 2004), Persian (Sadrzadeh 2007), Arabic (Bargelli and Lambek 2001b), Japanese (Cardinal 2002), and Latin (Casadio and Lambek 2005).

The distributional part assigns meanings to individual words by associating to them, for example, statistical co-occurrence vectors (Mitchell and Lapata 2008). The DisCoCat model is thus a way of interpreting compositions of meanings via grammatical structure.

In this article we study the notion of translating between compositional distributional models of language by analysing translation from Japanese into English. On the compositional side, a translation is a strong monoidal functor. It is easy to demonstrate that such a functor is too rigid to handle the translation of even simple phrases between languages which have different word order. We show that one can keep using the gadget of monoidal functors as long as the underlying pregroup grammars are decorated with additional structure.

We begin by introducing basic notions about the compact closed categories we work with – namely pregroups and finitely generated vector spaces – and define our notion of translation functor. Next, we give an introduction to basic Japanese grammar and the pregroup structure we use to model it. Finally, we introduce notions of pregroup decorations and use them to give a structured framework for translating Japanese sentences.

2

THEORETICAL BACKGROUND

2.1

Compact closed structures

The key to the DisCoCat model is that both the category of pregroups and the category of finitely generated vector spaces are *compact closed*

categories. This allows for compositional characteristics of grammar to be incorporated into the distributional spaces of meaning.

For completeness, we provide here a definition of compact closure. The reader is encouraged to consult (Kelly and Laplaza 1980) for a more complete and technical reference.

DEFINITION 1 A compact closed category is a category \mathcal{C} together with a bifunctor

$$- \otimes - : \mathcal{C} \times \mathcal{C} \rightarrow \mathcal{C},$$

called tensor product, which is associative up to natural isomorphism and possesses a two-sided identity element I , and each object $A \in \mathcal{C}$ has a right dual A^r and a left dual A^ℓ with the following morphisms

$$A \otimes A^r \xrightarrow{\varepsilon_A^r} I \xrightarrow{\eta_A^r} A^r \otimes A,$$

$$A^\ell \otimes A \xrightarrow{\varepsilon_A^\ell} I \xrightarrow{\eta_A^\ell} A \otimes A^\ell.$$

Moreover, the ε and η maps satisfy the “yanking” conditions:

$$\begin{aligned} (1_A \otimes \varepsilon_A^\ell) \circ (\eta_A^\ell \otimes 1_A) &= 1_A & (\varepsilon_A^r \otimes 1_A) \circ (1_A \otimes \eta_A^r) &= 1_A \\ (\varepsilon_A^\ell \otimes 1_A) \circ (1_{A^\ell} \otimes \eta_A^\ell) &= 1_{A^\ell} & (1_{A^r} \otimes \varepsilon_A^r) \circ (\eta_A^r \otimes 1_{A^r}) &= 1_{A^r}. \end{aligned}$$

The upshot of compact closure is that we want to have elements which “cancel each other out” and we can decompose the identity into a product.

Recalling pregroups

2.2

DEFINITION 2 A pregroup is a tuple $(P, \cdot, 1, -^\ell, -^r, \leq)$ where $(P, \cdot, 1, \leq)$ is a partially ordered monoid and the unary operations $-^\ell, -^r$ (the left and the right dual) satisfy for all $x \in P$ the following relations:

$$\begin{aligned} x \cdot x^r &\leq 1 & x^\ell \cdot x &\leq 1 \\ 1 &\leq x^r \cdot x & 1 &\leq x \cdot x^\ell. \end{aligned}$$

The operation sign \cdot is omitted unless it is relevant. It is immediate to check that the following relations hold in every pregroup:

$$\begin{aligned} 1^\ell &= 1 = 1^r & (x^\ell)^r &= x = (x^r)^\ell \\ (xy)^\ell &= y^\ell x^\ell \text{ and } (xy)^r = y^r x^r & \text{if } x \leq y &\text{ then } y^\ell \leq x^\ell \\ & & &\text{and } y^r \leq x^r. \end{aligned}$$

We model the grammar of a natural language by freely generating a pregroup from a set of grammatical types. Each word in the dictionary is assigned an element of the pregroup which corresponds to its linguistic function, e.g. noun, verb, adjective, etc. A string of words is interpreted by multiplying the elements assigned to the constituent parts in syntactic order. If a string of words satisfies the relation $w_1 w_2 \dots w_n \leq s$ we say that the string *reduces* to the type s .

EXAMPLE 1 Suppose there are two grammatical types: noun n and sentence s . Grammar is modelled as the free pregroup $\text{PGrp}(\{n, s\})$. Consider the sentence *Pigeons eat bread*. We assign the type n to *pigeons* and *bread* and the type $n^r s n^\ell$ to the transitive verb *eat*. The sentence overall has type $n(n^r s n^\ell)n$ and the following reductions hold:

$$n(n^r s n^\ell)n = (nn^r)s(n^\ell n) \leq (1)s(n^\ell n) \leq s(1) \leq s.$$

In this case we say that *Pigeons eat bread* is a well-formed sentence since in the pregroup $\text{PGrp}(\{n, s\})$ the phrase reduces to the correct type.

The two individual reductions could have been performed in a different order. Lambek's *Switching Lemma* (Lambek 1997, Proposition 2) tells us that in any computation performed in a freely generated pregroup, we may assume without loss of generality that all contractions precede all expansions.

A pregroup can be viewed as a compact closed category. The objects of the category are the elements of the pregroup. There is an arrow $x \rightarrow y$ if and only if $x \leq y$, and the tensor product is given by the pregroup operation: $x \otimes y = xy$. The morphisms $\varepsilon^r, \varepsilon^\ell, \eta^r, \eta^\ell$ are defined in the obvious way. In terms of the ε and η maps, the reductions in this example can be represented as:

$$(\varepsilon_n^\ell \otimes 1_s \otimes \varepsilon_n^\ell)(n \otimes (n^r \otimes s \otimes n^\ell) \otimes n) \rightarrow s.$$

2.3

Meaning space

We encode the semantic structure of a natural language into the category of finitely generated vector spaces, which we denote by FVect . The arrows are linear transformations, and there is a natural monoidal structure given by the linear algebraic tensor product with unit \mathbb{R} ,

which also happens to be symmetric: $V \otimes W \simeq W \otimes V$. This implies that $V^\ell \simeq V^r \simeq V^*$, where the latter denotes the dual vector space.

Fixing a basis $\{v_i\}$ for the vector space V we get moreover that $V \simeq V^*$ and the structure morphisms of compact closure are given by

$$\begin{aligned} \varepsilon_V &= \varepsilon_V^r = \varepsilon_V^\ell : V^* \otimes V \rightarrow \mathbb{R} \\ &\text{where } \sum_{i,j} a_{ij} v_i \otimes v_j \mapsto \sum_{i,j} a_{ij} \langle v_i | v_j \rangle \\ \eta_V &= \eta_V^r = \eta_V^\ell : \mathbb{R} \rightarrow V \otimes V^* \\ &\text{where } 1 \mapsto \sum_i v_i \otimes v_i \text{ extended linearly.} \end{aligned}$$

If we denote by P both the pregroup and the corresponding category, the bridge between grammar and semantics is given by a strong monoidal functor

$$F : P \rightarrow \text{FVect},$$

which we call a *functorial language model*. The functor assigns vector spaces to atomic types: $F(1) = I$, $F(n) = N$ (the vector space of nouns), $F(s) = S$ (the vector space of sentences), etc. For words in P , monoidality tells us that $F(x \otimes y) = F(x) \otimes F(y)$. The compact closure is also preserved: $F(x^\ell) = F(x^r) = F(x)^*$. For example, we can interpret the transitive verb *eat* with type $n^r s n^\ell$ as a vector in

$$\begin{aligned} F(n^r \otimes s \otimes n^\ell) &= F(n^r) \otimes F(s) \otimes F(n^\ell) \\ &= F(n)^* \otimes F(s) \otimes F(n)^* = N \otimes S \otimes N. \end{aligned}$$

Pregroup reductions in P can be interpreted as semantic reductions in FVect using the corresponding ε and η maps. The reductions associated to a transitive verb are then given by

$$\begin{aligned} F(\varepsilon_n^r \otimes 1_s \otimes \varepsilon_n^\ell) &= F(\varepsilon_n^r) \otimes F(1_s) \otimes F(\varepsilon_n^\ell) \\ &= F(\varepsilon_n)^* \otimes F(1_s) \otimes F(\varepsilon_n)^* = \varepsilon_N \otimes 1_S \otimes \varepsilon_N. \end{aligned}$$

The meaning of a sentence or phrase is derived by interpreting the pregroup reduction as the corresponding semantic reduction of the tensor product of distributional meanings of individual words in the phrase. The previous example *Pigeons eat bread* is interpreted as

$$F(\varepsilon_n^r \otimes 1_s \otimes \varepsilon_n^\ell)(\text{Pigeons} \otimes \text{eat} \otimes \text{bread}).$$

2.4

Translating between functorial language models

Bradley *et al.* (2018) formalised the notion of a translation between functorial language models. We illustrate this construction with an example on translating simple noun phrases and the problems one may encounter.

DEFINITION 3 *Let $(\mathcal{C}, \otimes, 1_{\mathcal{C}})$ and $(\mathcal{D}, \odot, 1_{\mathcal{D}})$ be monoidal categories. A monoidal functor $F : \mathcal{C} \rightarrow \mathcal{D}$ is a functor equipped with a natural isomorphism $\Phi_{x,y} : F(x) \odot F(y) \rightarrow F(x \otimes y)$ for every pair of objects $x, y \in \mathcal{C}$ and an isomorphism $\phi : 1_{\mathcal{D}} \rightarrow F(1_{\mathcal{C}})$ such that for any triple of objects $x, y, z \in \mathcal{C}$, the following diagram commutes*

$$\begin{array}{ccc} (F(x) \odot F(y)) \odot F(z) & \xrightarrow{\Phi_{x,y} \odot 1_{F(z)}} & F(x \otimes y) \odot F(z) & \xrightarrow{\Phi_{x \otimes y, z}} & F((x \otimes y) \otimes z) \\ \downarrow & & \downarrow & & \downarrow \\ F(x) \odot (F(y) \odot F(z)) & \xrightarrow{1_{F(x)} \odot \Phi_{y,z}} & F(x) \odot F(y \otimes z) & \xrightarrow{\Phi_{x, y \otimes z}} & F(x \otimes (y \otimes z)) \end{array}$$

where the vertical arrows apply the associativity in their respective categories. Moreover, for every object $x \in \mathcal{C}$, the following two squares commute:

$$\begin{array}{ccc} 1_{\mathcal{D}} \odot F(x) & \longrightarrow & F(x) & & F(x) \odot 1_{\mathcal{D}} & \longrightarrow & F(x) \\ \downarrow & & \uparrow & & \downarrow & & \uparrow \\ F(1_{\mathcal{C}}) \odot F(x) & \longrightarrow & F(1_{\mathcal{C}} \otimes x) & & F(x) \odot F(1_{\mathcal{C}}) & \longrightarrow & F(x \otimes 1_{\mathcal{C}}). \end{array}$$

DEFINITION 4 *Let (F, Φ, ϕ) and (G, Ψ, ψ) be monoidal functors between the monoidal categories \mathcal{C} and \mathcal{D} . A monoidal natural transformation $\alpha : F \Rightarrow G$ is a natural transformation where the following diagrams commute:*

$$\begin{array}{ccc} F(x) \odot F(y) & \xrightarrow{\alpha(x) \odot \alpha(y)} & G(x) \odot G(y) & & 1_{\mathcal{D}} & & \\ \downarrow \Phi_{x,y} & & \downarrow \Psi_{x,y} & & \searrow \psi & & \\ F(x \otimes y) & \xrightarrow{\alpha_{x \otimes y}} & G(x \otimes y) & & F(1_{\mathcal{C}}) & \xrightarrow{\alpha(1_{\mathcal{C}})} & G(1_{\mathcal{C}}). \end{array}$$

DEFINITION 5 *Let $\mathcal{A} : P \rightarrow \text{FVect}$ and $\mathcal{B} : Q \rightarrow \text{FVect}$ be two functorial language models. A translation from F to G is a tuple (T, α) , where $T : P \rightarrow Q$ is a monoidal functor and $\alpha : \mathcal{A} \Rightarrow \mathcal{B} \circ T$ is a monoidal natural transformation.*

EXAMPLE 2 We attempt to translate simple phrases of the type *adjective + noun* from Japanese to English. We work on a restricted model. Let $J = \text{PGrp}(\{s_J, n_J\})$ be the free pregroup (or category) generated by the sentence and noun types in Japanese and let $E = \text{PGrp}(\{s_E, n_E\})$ be the free pregroup generated by the sentence and noun types in English.

The functorial language models are denoted by $\mathcal{J} : J \rightarrow \text{FVect}$ and $\mathcal{E} : E \rightarrow \text{FVect}$, respectively. The semantic assignment is straightforward: $\mathcal{J}(n_J) = N_J$, $\mathcal{J}(a_J) = A_J$, $\mathcal{E}(n_E) = N_E$, $\mathcal{E}(a_E) = A_E$.

The translation will consist of the monoidal functor $T : J \rightarrow E$, which sends $s_J \mapsto s_E$ and $n_J \mapsto n_E$. Automatically, we have that the type reduction is preserved in the corresponding languages, i.e. $T((n_J n_J^\ell) n_J) = T(n_J) = n_E$. Due to monoidality, it suffices to define the components $\alpha_{n_J}, \alpha_{s_J}$ of the natural transformation $\alpha : \mathcal{J} \Rightarrow \mathcal{E} \circ T$ in order to parse semantics.

Additionally, the natural transformation α must commute with the monoidal functor T . Pictorially, we have a commutative square:

$$\begin{array}{ccc} (N_J \otimes N_J) \otimes N_J & \xrightarrow{\mathcal{J}(\varepsilon_{N_J}^\ell \otimes 1_{\mathcal{J}})} & N_J \\ \downarrow \alpha_{(n_J n_J^\ell) n_J} & & \downarrow \alpha_{n_J} \\ (N_E \otimes N_E) \otimes N_E & \xrightarrow{\mathcal{E}(\varepsilon^\ell \otimes 1_{\mathcal{E}})} & N_E \end{array}$$

Consider the concrete words $\mathbf{red} \in N_E \otimes N_E$, $\mathbf{cat} \in N_E$, $\mathbf{akai} \in N_J \otimes N_J$, and $\mathbf{neko} \in N_J$. The diagram says that if we first use Japanese grammar rules to reduce $\mathbf{akai} \otimes \mathbf{neko}$ to $\mathbf{akai neko}$ and then translate to $\mathbf{red cat}$ is the same thing as first translating component-wise $\mathbf{akai} \otimes \mathbf{neko}$ to $\mathbf{red} \otimes \mathbf{cat}$ and then using English grammar rules to reduce to $\mathbf{red cat}$.

Since there is no discrepancy in word order, this example of phrasal translation works in the desired way. If we instead wanted to translate the phrase *akai neko* from Japanese into *pisică roșie* in Romanian, we would encounter some difficulties. The latter is a “noun + adjective” phrase, as the natural word order in Romanian for such phrases is the opposite to the word order in Japanese.

The reduction rule in Romanian is given by

$$n_R(n_R^r n_R) \mapsto n_R.$$

Suppose there exists a monoidal functor $T' : J \rightarrow R$ that takes Japanese grammar types to Romanian grammar types. Then we want to preserve the reduction rules, i.e.

$$T'((n_J n_J^\ell) n_J) = n_R(n_R^r n_R).$$

We thus obtain the condition: $T'(n_J^\ell) = n_R^r$. However, left and right adjoints must be preserved by a strong monoidal functor. Hence this condition cannot be fulfilled.

Section 4 introduces techniques that can help us overcome such problems with word order changes.

3 JAPANESE CRASH COURSE

3.1 Generalities

Japanese is a synthetic and agglutinative language. The usual word order is subject-object-verb (SOV) with topic-comment sentence structure. There are no definite/indefinite articles. Nouns possess neither grammatical gender nor number. Verbs and adjectives are conjugated for tense, voice, and aspect, but not person or number. Particles are attached to words to identify their grammatical role. We write sentences natively and employ the *Nihon-siki* romanisation system.

The sentence *The cat eats fish* can be represented in two different but closely related ways.

- (1) 猫 が 魚 を 食べる
 neko ga sakana wo taberu
 cat NOM fish ACC eat
- (2) 猫 は 魚 を 食べる
 neko ha sakana wo taberu
 cat TOP fish ACC eat

Note the use of the subject particle *ga*, the topic particle *ha*, and the direct object particle *wo*. Remark that Japanese distinguishes between topic and subject. The topic generally needs to be explicitly introduced at the beginning of a discourse, but as the discourse carries

on, the topic need not be the grammatical subject of every sentence. Both sentences translate into English as *The cat eats fish*, or *Cats eat fish*. However, a more pertinent interpretation of the second sentence is *As for the cat/Speaking of the cat, it eats fish*.

Another important aspect of word order in Japanese is head finality. Phrases can be broadly described as consisting of a **head** and a *modifier*. English is generally a head initial language. Consider for example the phrases: *to school*, *in England*, and *red cat*. The word that gets modified tends to come before the modifiers, the main exception being that nouns succeed the adjectives that modify them. In contrast, Japanese is a canonical example of a head final language. Our example phrases become

(3) 学校 へ
gakkō he
school to

(4) イギリス に
igirisu ni
England in

(5) 赤い 猫
akai neko
red cat

Head finality is also encountered in the case of relative clauses, which usually occur before the part of speech they modify. This phenomenon is demonstrated by the following pair of phrases.

(6) 女 が 赤い ワンピース を 着た
onna ga akai wanpîsu wo kita
woman NOM red dress ACC wore
'The woman wore a red dress'

(7) 赤い ワンピース を 着た 女
akai wanpîsu wo kita onna
red dress ACC wore woman
'The woman, who wore a red dress'

This is a prime example of a structure where the word order is changed during translation. The following section will develop the algebraic machinery to interpret such translations.

Subjects are habitually dropped when they are clear from context, and personal pronouns are used sparingly. We conclude this section with an example, which demonstrates how a very common reflexive/personal pronoun *zibun* ‘oneself’ can lead to ambiguous interpretations. *Zibun* is often used as a way for the speaker to refer either to themselves or to their interlocutor. The sentence

- (8) 自分 が 嘘つき か
 zibun ga usotuki ka
 oneself NOM liar QUESTION

can be translated as either *Am I a liar?* or *Are you a liar?* in the absence of further context.

3.2

Compositional model

Define $J = \text{PGrp}(\{\pi, n, s_1, s_2, s, o_1, \dots\})$ to be the pregroup of grammar types associated to Japanese. Following Cardinal 2002 with slight modifications, we define the following atomic types:

- π pronoun,
- n noun,
- s_1, s_2 imperfective / perfective sentence,
- \bar{s} topicalised sentence,
- s sentence,
- o_1 nominative case,
- o_2 accusative case,
- o_3 dative case,
- o_4 genitive case,
- o_5 locative case,
- o_6 lative case,
- o_7 ablative case,
- etc.

We also impose the following reductions in J :

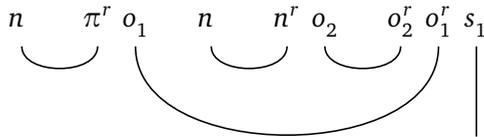
$$s_i \rightarrow s \qquad \bar{s} \rightarrow s \qquad n \rightarrow \pi.$$

We now discuss how to assign types to various parts of speech. Revisiting the example sentence *neko ga sakana wo taberu* ‘the cat eats

fish', the words *neko* and *sakana* are both nouns and thus have type n . The subject particle *ga* has type $\pi^r o_1$, the direct object particle *wo* has type $n^r o_2$ and the transitive verb *taberu* then has type $o_2^r o_1^r s_1$. The sentence then has type $n(\pi^r o_1)n(n^r o_2)(o_2^r o_1^r s_1)$ and we can derive the following type reductions:

$$\begin{aligned}
 n(\pi^r o_1)n(n^r o_2)(o_2^r o_1^r s_1) &\rightarrow (n\pi^r)o_1(\pi\pi^r)o_2(o_2^r o_1^r s_1) \\
 &\rightarrow (\pi\pi^r)o_1(\pi\pi^r)o_2(o_2^r o_1^r s_1) \\
 &\rightarrow o_1 o_2(o_2^r o_1^r s_1) \\
 &\rightarrow o_1(o_2 o_2^r)o_1^r s_1 \\
 &\rightarrow o_1 o_1^r s_1 \\
 &\rightarrow s_1 \\
 &\rightarrow s
 \end{aligned}$$

to see that the sentence is well-formed and reduces to the correct grammatical type. Here we used the reductions $n \rightarrow \pi$ and $s_1 \rightarrow s$ together with different applications of the contraction morphism ε . Graphically, this type reduction can be seen in the following diagram, where a lower bracket indicates that a contraction morphism of the type ε was applied.



Since word order is flexible, the same sentence could have been written as *sakana wo neko ga taberu*, and then *taberu* would have been assigned the type $o_1^r o_2^r s_1$. As we want to take advantage of the *Switching Lemma* while performing computations, we want to restrict ourselves to working with freely generated pregroups. Situations where certain words or verbs can be assigned different types are generally handled by adding *metarules*. Informally, a metarule stipulates that if a grammar contains rules that match a specified pattern, then it also contains rules that match some other specified pattern. In our concrete example, we could impose the following metarule.

METARULE 1 Any transitive verb that has type $o_1^r o_2^r s_i$ also has type $o_2^r o_1^r s_i$.

Moving away from transitive verbs, the ablative particle *kara* has type $\pi^r o_7$ and the lative particle *he* has type $\pi^r o_6$. In the following example, the verb *untensita* has type $o_6^r o_7^r s_2$.

- (9) 家 から 駅 へ 運転した
 ie kara eki he untensita
 house ABL station LAT drove
 ‘(I) drove from home to the train station.’

Causative passive verbs take a subject and an indirect object marked with the dative particle *ni* of type $\pi^r o_3$. For instance, the verb *yomaseta* ‘x made y read’ has type $o_2^r o_3^r o_1^r s_2$.

- (10) 先生 が 私 に 本 を 読ませた
 sensei ga watasi ni hon wo yomaseta
 teacher NOM I DAT book ACC read-CAUSE-PAS
 ‘The teacher made me read the book.’

The genitive particle *no* has type $\pi^r o_4$ together with a metarule that states that type o_4 is equivalent to type nn^ℓ . The possessor is always on the left in a genitive construction. The topic particle *ha* is distinguished from the subject particle *ga* and has type $\pi^r \bar{s}s^\ell$, i.e. *ha* requires a topic on the left and a sentence about the topic on the right.

- (11) 私 の 車 は 箸 を 渡れない
 watasi no kuruma ha hasi wo watarenai
 I GEN car TOP bridge ACC cross-POT-NEG
 ‘I cannot cross the bridge with my car/About my car, it cannot cross the bridge.’

In the latter example, the type reduction goes as follows:

$$\begin{aligned}
 & \pi(\pi^r o_4)n(\pi^r \bar{s}s^\ell)n(\pi^r o_2)(o_2^r s_1) \\
 \rightarrow & (\pi\pi^r)o_4n(\pi^r \bar{s}s^\ell)(n\pi^r)(o_2 o_2^r)s && \text{associativity} \\
 \rightarrow & (1)(nn^\ell)n(\pi^r \bar{s}s^\ell)(n\pi^r)(1)s && \text{contractions + genitive} \\
 & && \text{metarule} \\
 \rightarrow & n(n^\ell n)(\pi^r \bar{s}s^\ell)(\pi\pi^r)s && n \rightarrow \pi \\
 \rightarrow & (n\pi^r)\bar{s}(s^\ell s) && \text{associativity} \\
 \rightarrow & \bar{s} && \text{contractions} \\
 \rightarrow & s.
 \end{aligned}$$

TRANSLATION AND DECORATED
PREGROUPS

4

Decorated pregroups

4.1

As Example 2 shows, our initial machinery is not suited to translating phrases between languages with different word orders. The morphism of pregroups (or monoidal functor) $T : P \rightarrow Q$ that transfers information from the source language to the target language happens to be too rigid. We decorate pregroups with additional structures so that we can have more control over the monoid's operation. To this end, we define anti-homomorphisms for the purpose of inverting word order and pregroups with braces and β -pregroups to get more refined control over associativity.

DEFINITION 6 *An anti-homomorphism of monoids is a map $\Phi : P \rightarrow Q$ such that for all elements $x, y \in P$ we have $\Phi(xy) = \Phi(y)\Phi(x)$.*

DEFINITION 7 *Let (P, \cdot) be a monoid. The opposite monoid $(P^{op}, *)$ is the monoid which has the same elements as P and the operation for all $x, y \in P^{op}$ is given by $x * y = y \cdot x$. Observe that $(P, \cdot) \simeq (P^{op}, *)$.*

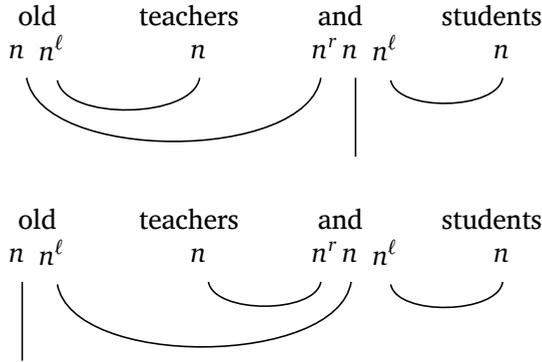
In light of this, an anti-homomorphism can be viewed as a morphism from the opposite monoid $\Phi : P^{op} \rightarrow Q$. Additionally, an anti-homomorphism of pregroups takes left adjoints to right adjoints and vice-versa.

EXAMPLE 3 In Example 2 the problem of translating “adjective + noun” phrases from Japanese into Romanian can be solved by setting the translation functor to be an anti-homomorphism that sends $n_J \mapsto n_R$. Then the functor T preserves the desired reductions

$$T((n_J n_J^\ell) n_J) = T(n_J) (T(n_J^\ell) T(n_J)) = n_R (n_R^r n_R) \rightarrow n_R.$$

Parsing longer phrases and full sentences adds new layers of complexity. For instance, in simple short phrases there often is exactly one way of performing type reductions in order to assess the syntactic type of a phrase. Associativity can introduce ambiguity while parsing phrases. The following example demonstrates this.

EXAMPLE 4 Consider the phrase *old teachers and students*. We assign type n to *teachers* and *students*. We assign the type nn^ℓ to the adjective *old*. The conjunction *and* in this phrase requires two inputs of noun type to produce a noun phrase and is thus assigned $n^r n^\ell$. We can use the associativity of the monoid operation to perform two distinct type reductions.



Both type reductions give the desired noun phrase. However, the two interpretations are slightly different. The first one attributes the adjective *old* to *teachers* only, and so the sentence is parsed as *(old teachers) and students*, while the second type reduction attributes *old* to both *teachers* and *students*, giving the phrase *old (teachers and students)*.

One can construct examples where changing the order of reductions can make the difference between reducing down to a well-formed sentence and reducing down to a phrase that cannot be grammatically accepted. For this reason, one can add a modality or a β -structure to the pregroup to locally suppress associativity. This is to ensure that our phrases reduce to the correct type or that we distribute modifiers in a prescribed way.

Pregroups with modalities were first introduced in Fadda 2002 and their logic was more extensively studied in Kiślak-Malinowska 2007.

DEFINITION 8 A β -pregroup is a pregroup $(P, \cdot, 1, -^\ell, -^r, \leq)$ together with a monotone mapping $\beta : P \rightarrow P$ such that β has a right adjoint $\hat{\beta} : P \rightarrow P$, i.e. for all $x, y \in P$ we have $\beta(x) \leq y$ if and only if $x \leq \hat{\beta}(y)$.

In practice, we enrich our pregroup grammars with types with modalities to indicate certain reductions must be performed first.

EXAMPLE 5 In our previous example, we can prescribe the parsing (*old teachers*) and *students* by assigning the types

$$n[\beta(n)]^\ell \cdot [\beta(n)] \cdot n^r nn^\ell \cdot n$$

and the parsing *old (teachers and students)* by assigning the types

$$nn^\ell \cdot [\beta(n)] \cdot [\beta(n)]^r nn^\ell \cdot n.$$

We now have ways to invert word order globally and block associativity locally. We conclude this section by introducing a new type of decoration which allows us to locally control word order.

The reader is also encouraged to consult Stabler 2008 for an introduction to tupled pregroups and Lambek 2010 for an analysis of French sentences using products of pregroups.

Next, we introduce a new pregroup decoration.

DEFINITION 9 A monoid with k -braces $(P, \cdot, 1)$ is a free monoid in which every word is a prescribed concatenation of $k > 0$ distinguished subwords. Extending this and subsequent definitions to pregroups with k -braces is immediate.

EXAMPLE 6 Consider the free monoid on two letters

$$F = \text{Mon}(\{a, b\}).$$

Viewing F as a monoid with 2-braces, $\langle abba \rangle \langle b \rangle$ and $\langle abb \rangle \langle ab \rangle$ are distinct words because they have distinct distinguished subwords.

DEFINITION 10 A morphism of monoids with k -braces $f : (P, \cdot) \rightarrow (Q, *)$ is a morphism of monoids $f : (P, \cdot) \rightarrow (Q, *)$ which preserves distinguished subwords. In symbols:

$$f(\langle w_1 \rangle \cdot \dots \cdot \langle w_k \rangle) = \langle f(w_1) \rangle * \langle f(w_2) \rangle * \dots * \langle f(w_k) \rangle.$$

EXAMPLE 7 [Some useful constructions] We define two morphisms of monoids with braces which are useful in understanding translations.

First, let P be a monoid with 2-braces and consider a word

$$w = \langle w_1 \rangle \langle w_2 \rangle.$$

Since the underlying monoid of P is free, we can view w as an element of the free product $P * P \simeq P$ where the distinguished subword w_i

belongs to the i -th factor. Take the following sequence of monoid morphisms

$$\begin{aligned} \Psi : P \simeq P * P &\xrightarrow{f} P \times P \xrightarrow{g} P^{\text{op}} \times P^{\text{op}} \xrightarrow{h} \\ &\xrightarrow{h} P^{\text{op}} * P^{\text{op}} \xrightarrow{i} Q \end{aligned}$$

Here, f is the canonical surjection sending $\langle w_1 \rangle \langle w_2 \rangle \mapsto (w_1, w_2)$, g is a pair of anti-isomorphisms which act like the identity on atomic types $(w_1, w_2) \mapsto (w_1^{\text{op}}, w_2^{\text{op}})$, h is the canonical injection sending $(w_1^{\text{op}}, w_2^{\text{op}}) \mapsto \langle w_1^{\text{op}} \rangle \langle w_2^{\text{op}} \rangle$ and i is some fixed homomorphism of monoids with 2-braces.

Secondly, let P be a monoid with 3-braces. We construct in a similar fashion the following morphism.

$$\begin{aligned} \Xi : P \simeq P * P * P &\longrightarrow P \times P \times P \longrightarrow P \times P^{\text{op}} \times P \longrightarrow \\ &\longrightarrow P * P^{\text{op}} * P \longrightarrow P * P * P \simeq P \longrightarrow Q \end{aligned}$$

We now proceed with concrete examples of phrasal translations. Throughout the remainder of the section, we work with two functorial language models: $\mathcal{J} : J \rightarrow \text{FVect}$ for Japanese and $\mathcal{E} : E \rightarrow \text{FVect}$ for English. We also impose the following useful metarule.

METARULE 2 Any verb of type $so_1^r w$ also has type $o_1^l sw$, where w stands for all the remaining required complements.

4.2

There is/There exists

Japanese has two verbs of existence, *iru* and *aru*, which are used for animate and inanimate beings, respectively. They both roughly mean ‘to be’, although a more common English translation is ‘there is/there exists’.

Consider the following sentence.

- (12) 森 に 猫 が いる
 mori ni neko ga iru
 forest LOC cat NOM be

A human translator has numerous ways of approaching this sentence. A standard and literal SVO translation is *A cat is in the forest*. An easy SVO upgrade would be *A cat lives in the forest*. Considering

that this is a short story meant for children, one could even opt for *In the forest lives a cat* to induce a fairy tale type atmosphere to the text. In this article, we choose to translate this using a straightforward anti-homomorphism and thus we aim for *There is a cat in the forest*.

We work with the following reduced models for grammar:

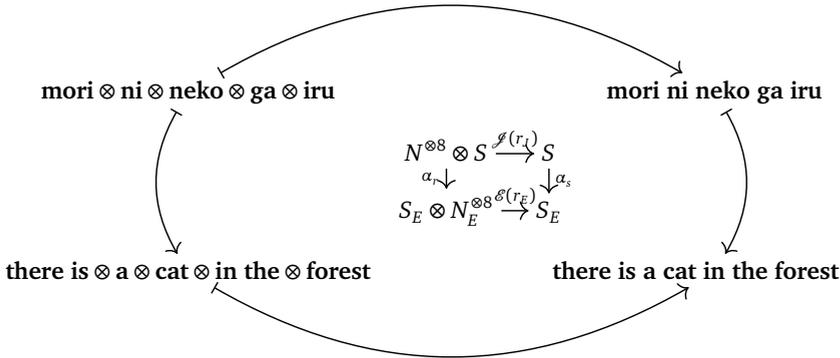
$$J = \text{PGrp}(\{n, o_1, o_5, s\}) \text{ and } E = \text{PGrp}(\{n_E, o_{1E}, o_{5E}, s_E\}).$$

The translation functor at the level of syntax is given by the anti-homomorphism $T : J \rightarrow E$ which sends $n \mapsto n_E, o_1 \mapsto o_{1E}, o_5 \mapsto o_{5E}, s \mapsto s_E$. At the level of semantics we have $F(n) = F(o_1) = F(o_5) = N, F(s) = S$ and $G(n_E) = G(o_{1E}) = G(o_{5E}) = N_E, G(s_E) = S_E$.

In J we have the type reduction $r = n(n^r o_5)n(n^r o_1)(o_1^r o_5^r s) \leq s$. After applying the translation functor T we get:

$$\begin{aligned} & T(n(n^r o_5)n(n^r o_1)(o_1^r o_5^r s)) \\ &= T(s)T(o_5^r)T(o_1^r)T(o_1)T(n^r)T(n)T(o_5)T(n^r)T(n) \\ &= (s_E o_{5E}^l o_{1E}^l) o_{1E} (n_E^l n_E) o_{5E} (n_E^l n_E) \\ &\rightarrow s_E o_{5E}^l (o_{1E}^l o_{1E}) o_{5E} \\ &\rightarrow s_E (o_{5E}^l o_{5E}) \\ &\rightarrow s_E. \end{aligned}$$

At the level of semantics we define the natural transformation $\alpha : \mathcal{J} \Rightarrow \mathcal{E} \circ T$ to act in the expected way, i.e. the map $N \rightarrow N_E$ sends **neko** \mapsto **cat**, **mori** \mapsto **forest** and the map $S \rightarrow S_E$ sends **iru** \mapsto **there is**. We also impose **ga** \mapsto **a** and **ni** \mapsto **in the**. The commutativity of the following diagram is immediate.



4.3

Simple SOV sentences

We describe a procedure for translating the following sentence.

- (13) 医者 は 手紙 を 書く
 issya ga tegami wo kaku
 doctor NOM letter ACC write
 ‘The doctor writes a letter.’

We work with the grammars

$$J = \text{PGrp}(\{n, o_1, o_2, s\}) \text{ and } E = \text{PGrp}(\{n_E, o_{1E}, o_{2E}, s_E\}).$$

The words *issya* and *tegami* are assigned the noun type n , the particles *ga* and *wo* have the usual types $n^r o_1$ and $n^r o_2$, respectively, and the transitive verb *kaku* has type $o_2^r o_1^r s$. The sentence is clearly well-formed: $n(n^r o_1)n(n^r o_2)(o_2^r o_1^r s) \rightarrow s$.

Here we employ the notion of a pregroup with 2-braces. In principle, for an SOV sentence we assign braces as follows: $\langle S \rangle \langle OV \rangle$. In our particular sentence, this becomes

$$\langle n(n^r o_1) \rangle \langle n(n^r o_2)(o_2^r o_1^r s) \rangle.$$

We define our translation functor Ψ to be the morphism of monoids with braces defined in Example 7. Together with Metarule 2 this gives:

$$\begin{aligned} \Psi \langle n(n^r o_1) \rangle \langle n(n^r o_2)(o_2^r o_1^r s) \rangle &= \langle (o_{1E} n_E^\ell) n_E \rangle \langle (s_E o_{1E}^\ell o_{2E}^\ell) (o_{2E} n_E^\ell) n \rangle \\ &= \langle (o_{1E} n_E^\ell) n_E \rangle \langle (o_{1E}^r s_E o_{2E}^\ell) (o_{2E} n_E^\ell) n \rangle \end{aligned}$$

Then α can be defined on atomic types as follows: **issya** \mapsto **doctor**, **tegami** \mapsto **letter**, **kaku** \mapsto **write**, and the translation (Ψ, α) gives

$$(A/\text{The}) \text{ doctor write}(s) \text{ (a/the) letter.}$$

Again, the articles and the conjugation of *write* into third person singular can either be added by brute force in our model by adding meanings to the particles *ga* and *wo*, or one can verify agreement and articles separately as a different step in the translation process.

Interpreting relative pronouns in various languages in terms of pregroups proves to be quite challenging. In Sadrzadeh *et al.* 2013 and Sadrzadeh *et al.* 2014, the authors add the additional structure of a Frobenius algebra on the pregroup. Informally, a Frobenius algebra structure enriches the ε, η functorial yoga with additional maps, the most important of which are called “copying map” and “uncopying map.” These new morphisms allow one to better keep track of information inside a phrase. For instance, in the English sentence

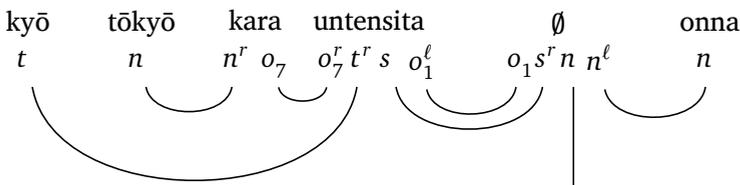
The woman, who drove from Tokyo today, was late to the party

the new morphisms can formalise the fact that the subject of the main clause *The woman was late to the party* and the subject of the relative clause *who drove from Tokyo today* are one and the same. The relative pronoun *who* acts as a bridge that “copies” the subject into the relative clause and then transfers it back into the main clause.

We translate the following relative clause.

- (14) 今日 東京 から 運転した 女
 kyō tōkyō kara untensita onna
 today Tokyo ABL drove woman
 ‘The woman who drove from Tokyo today.’

We assign types in a less straightforward way. We first insert an empty word between the modifier *tōkyō kara untensita* and the head *onna*. We assign the following types: *tōkyō* and *onna* are both type n , the ablative particle *kara* has type $n^r o_7$, *kyō* has type t (temporal adverb), the verb *untensita* has type $o_7^r t^r s o_1^\ell$ and the empty word acts like a phantom relative pronoun with type $o_1 s^r n n^\ell$.



This construction generates a noun phrase and it can be translated using a straightforward anti-homomorphism. The advantage of this underhanded construction is that now we can translate the empty word as the relative pronoun *who* or *that*. This ties in perfectly with the Frobenius algebra approach of Sadrzadeh *et al.* (2013). In this example we modelled our relative clause as what the authors of the reference call a subject relative clause.

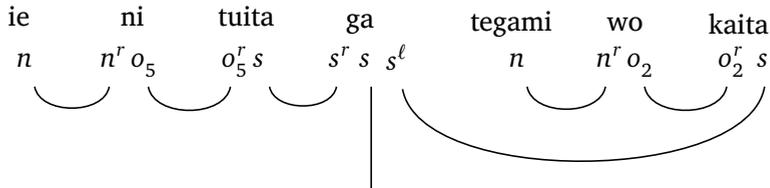
4.5

Coordinate sentences

The simplest way of coordinating sentences is by connecting them with the particle *ga* ‘and’ to which we assign the type $s^r ss^\ell$. We translate the following sentence where subjects are omitted.

- (15) 家 に 着いた が^s 手紙 を 書いた
 ie ni tuita ga tegami wo kaita
 house LOC arrived and letter ACC wrote
 ‘I arrived home and wrote a letter.’

In Japanese we have the following reduction diagram.



We decorate the pregroup with braces and assign the following type

$$\langle n \cdot n^r o_5 \cdot o_5^r s \rangle \langle s^r ss^\ell \rangle \langle n \cdot n^r o_2 \cdot o_2^r s \rangle.$$

Extending the morphism Ψ from Example 7 to monoids with 3-braces, we obtain

$$\begin{aligned} \Psi \langle n \cdot n^r o_5 \cdot o_5^r s \rangle \langle s^r ss^\ell \rangle \langle n \cdot n^r o_2 \cdot o_2^r s \rangle \\ = \langle s_E o_{5E}^\ell \cdot o_{5E} n_E^\ell \cdot n_E \rangle \langle s_E^r s_E s_E^\ell \rangle \langle s_E o_{2E}^\ell \cdot o_{2E} n_E^\ell \cdot n_E \rangle. \end{aligned}$$

Working under the assumption that an omitted subject refers to the first person singular, the translation, after applying a suitably defined α , is

(I) arrived home and (I) wrote (a) letter.

Putting it all together

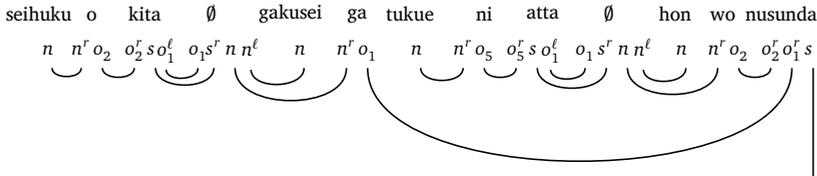
4.6

We combine all our techniques to study a more complex sentence.

(16) 制服 を 着た 学生 が 机 に あった
 seihuku o kita gakusei ga tukue ni atta
 uniform ACC wore student NOM desk LOC was
 本 を 盗んだ
 hon wo nusunda
 book ACC stole

‘The student, who wore a uniform, stole the book, which was on the desk.’

This is a standard SOV sentence, where both the subject and the direct object are modified by relative clauses. In the Japanese pregroup grammar, we have the following straightforward reductions.



One may observe that in the diagram above we use associativity to our advantage to prove that the sentence reduces to the correct syntactic type. To get a fail-safe reduction and translation we decorate our pregroup grammar with braces and a β -structure. The sentence is then assigned the type

$$\langle n \cdot n^r o_2 \cdot o_2^r o_1^l \cdot o_1 s^r n \beta(n^l) \cdot \beta(n) \cdot n^r o_1 \rangle$$

$$\langle n \cdot n^r o_5 \cdot o_5^r s o_1^l \cdot o_1 s^r n \beta(n^l) \cdot \beta(n) \cdot n^r o_2 \cdot o_2^r o_1^r s \rangle$$

and after applying the morphism Ψ from Example 7 together with Metarule 2, the sentence translates to

(A/The) student, who wore (a/the) uniform, stole (a/the) book, which was (LOC) desk.

4.7

A Farsi to Japanese example

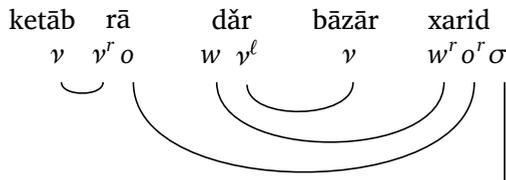
Farsi has certain similarities to Japanese which make translations (at the syntactic level, at least) somewhat simpler. For instance, Farsi also has SOV word order, nouns do not possess grammatical gender, and it is a pro-drop language. A key structural difference is that Farsi uses both prepositions and postpositons as case markers.

Following Sadrzadeh (2007), we use the following (reduced) pre-group to model Farsi grammar $F = \text{PGrp}(\{\nu, \sigma, o, w\})$, where atomic types represent nouns, sentences, direct objects, and prepositional phrases respectively. For Japanese, we use $J = \text{PGrp}(\{n, s, o_2, o_5\})$, with the usual meanings. Denote the two functorial language models as $\mathcal{F} : F \rightarrow \text{FVect}$ and $\mathcal{J} : J \rightarrow \text{FVect}$.

We are interested in translating the following sentence from Farsi to Japanese.

- (17) *ketāb rā dār bāzār xarid*
 book ACC PREP market bought
 ‘He/She bought a book from the market.’

Here *ketāb rā* is the direct object, *dār bāzār* is the prepositional phrase and *xarid* is the transitive verb in the past tense. This example sentence drops the subject and uses both a postposition *rā* and a preposition *dār* to mark cases. In Farsi, we have the following reduction.



The functorial language models \mathcal{F}, \mathcal{J} send $\nu, o, w \mapsto N_F$ (Farsi nouns) and $\sigma \mapsto S_F$ (Farsi sentences), and also $n, o_2, o_5 \mapsto N_J$ (Japanese nouns) and $s \mapsto S_J$ (Japanese sentences). The natural transformation $\alpha : \mathcal{F} \Rightarrow \mathcal{J} \circ i$ is given by **ketāb** \mapsto **hon**, **rā** \mapsto **o**, **dār** \mapsto **de**, **bāzār** \mapsto **itiba** and **xarid** \mapsto **kaimasita**. At the syntactic level, we define some monoidal translation functor $T : F \rightarrow J$ which takes $\nu \mapsto n, \sigma \mapsto s, o \mapsto o_2$, and $w \mapsto o_5$.

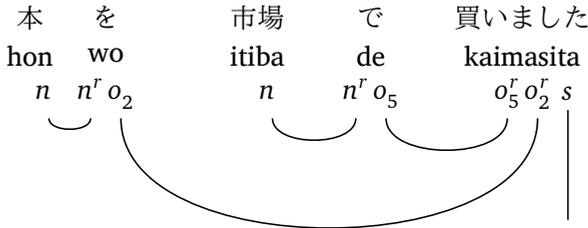
The pregroups are decorated with 3-braces. The sentence is assigned the type

$$\langle \nu \cdot \nu^r o \rangle \langle w \nu^\ell \cdot \nu \rangle \langle w^r o^r \sigma \rangle.$$

Syntactically, the translation functor is taken to be Ξ from Example 7. The word order is altered as follows

$$\Xi \langle \nu \cdot \nu^r o \rangle \langle w \nu^\ell \cdot \nu \rangle \langle w^r o^r \sigma \rangle = \langle n \cdot n^r o_2 \rangle \langle n \cdot n^r o_5 \rangle \langle o_5^r o_2^r s \rangle$$

which leads to the following type reduction in Japanese.



FUTURE WORK

5

In this article, we introduced decorated pregroups and used them as a means of constructing a compositional notion of translation between natural languages with different word order. The aim was to demonstrate that one can maintain a categorical approach to modelling translation without compromising on functoriality altogether. Some of our constructions are ad-hoc and there is room for improving most of them.

First, there is the issue of translating between a language where nouns do not have grammatical gender and number to a language that does. Using product pregroups or tupled pregroups to handle grammatical agreement could be a way forward, although a straightforward model for achieving this appears elusive.

Secondly, one could study translations between languages which have more featural and structural differences. For example, how could we interpret (functorially) translations between a language which has

nominative-accusative alignment and a language that has ergative-absolutive (or split-ergative) alignment?

Thirdly, this article focuses heavily on syntax. It would be interesting to model how meaning in translation can be negotiated between different speakers and how one can keep track of their evolving semantic spaces. On a more technical note, one could change the meaning space from $FVect$ to a category that possesses more substantial structure such as $ConvexRel$, the category where the objects are *convex algebras* and the morphisms as *convex relations*. In Bolt *et al.* 2019 the authors showed that $ConvexRel$ is a compact closed symmetric monoidal category and is thus suitable for modelling semantics in a compositional distributional functorial language model.

Finally, separate from the question of translation, some attention could be dedicated to expanding the work of Cardinal (2002; 2006; 2007) and producing a more complete pregroup approach to analysing other aspects of grammar that are typical to Japanese. In particular, the structure of coordinate and subordinate sentences and internally-headed relative clauses are of particular interest to the author.

REFERENCES

- Daniele BARGELLI and Joachim LAMBEK (2001a), An algebraic approach to French sentence structure, in *International Conference on Logical Aspects of Computational Linguistics*, pp. 62–78, Springer, DOI:10.1007/3-540-48199-0_4.
- Donna BARGELLI and Joachim LAMBEK (2001b), An algebraic approach to Arabic sentence structure, *Linguistic Analysis*, 31(3):301–315, Corpus ID: 56575156.
- Joe BOLT, Bob COECKE, Fabrizio GENOVESE, Martha LEWIS, Dan MARSDEN, and Robin PIEDELEU (2019), *Interacting conceptual spaces I: Grammatical composition of concepts*, Springer, DOI:10.1007/978-3-030-12800-5_9.
- Tai-Danae BRADLEY, Martha LEWIS, Jade MASTER, and Brad THEILMAN (2018), Translating and evolving: Towards a model of language change in discocat, *ECPTS*, DOI:10.4204/EPTCS.283.4.
- Kumi CARDINAL (2002), *An algebraic study of Japanese grammar*, PhD Thesis.
- Kumi CARDINAL (2006), Type grammar meets Japanese particles, in *Proceedings of the 20th Pacific Asia Conference on Language, Information and Computation*, pp. 142–149.

- Kumi CARDINAL (2007), A pregroup analysis of Japanese causatives, in *Proceedings of the Korean Society for Language and Information Conference*, pp. 96–104, Korean Society for Language and Information.
- Claudia CASADIO and Jim LAMBEK (2005), A computational algebraic approach to Latin grammar, *Research on Language and Computation*, 3(1):45–60, DOI:10.1007/s11168-005-1286-0.
- Bob COECKE, Mehrnoosh SADRZADEH, and Stephen CLARK (2010), Mathematical foundations for a compositional distributional model of meaning, *Linguistic Analysis*, DOI:10.48550/arXiv.1003.4394.
- Mario FADDA (2002), Towards flexible pregroup grammars, *New Perspectives in Logic and Formal Linguistics*, pp. 95–112.
- Gregory M. KELLY and Miguel L. LAPLAZA (1980), Coherence for compact closed categories, *Journal of Pure and Applied Algebra*, 19:193–213, DOI:10.1016/0022-4049(80)90101-2.
- Aleksandra KIŚLAK-MALINOWSKA (2007), On the logic of β -pregroups, *Studia Logica*, 87(2):323–342, DOI:10.1007/s11225-007-9090-5.
- Joachim LAMBEK (1997), Type grammar revisited, in *International conference on logical aspects of computational linguistics*, pp. 1–27, Springer, DOI:10.1007/3-540-48975-4_1.
- Joachim LAMBEK (2010), Exploring feature agreement in French with parallel pregroup computations, *Journal of Logic, Language and Information*, 19(1):75–88, DOI:10.1007/s10849-009-9098-5.
- Joachim LAMBEK and Anne PRELLER (2004), An algebraic approach to the German sentence, *Linguistic Analysis*, HAL ID: irmm-00108541.
- Jeff MITCHELL and Mirella LAPATA (2008), Vector-based models of semantic composition, in *Proceedings of ACL-08: HLT*, pp. 236–244, <https://aclanthology.org/P08-1028.pdf>.
- Mehrnoosh SADRZADEH (2007), *Pregroup analysis of Persian sentences*, Citeseer, <https://www.cs.ox.ac.uk/files/2416/PersPreGroup.pdf>.
- Mehrnoosh SADRZADEH, Stephen CLARK, and Bob COECKE (2013), The Frobenius anatomy of word meanings I: subject and object relative pronouns, *Journal of Logic and Computation*, 23(6):1293–1317, DOI:10.1093/logcom/ext044.
- Mehrnoosh SADRZADEH, Stephen CLARK, and Bob COECKE (2014), The Frobenius anatomy of word meanings II: possessive relative pronouns, *Journal of Logic and Computation*, 26(2):785–815, DOI:10.1093/logcom/exu027.
- Edward P. STABLER (2008), Tupled pregroup grammars, *Computational Algebraic Approaches to Natural Language*, pp. 23–52.

Valentin Boboc

Valentin Boboc

© 0000-0003-0661-5404

valentinboboc@icloud.com

University of Manchester

Valentin Boboc (2023), *An algebraic approach to translating Japanese*, *Journal of Language Modelling*, 11(1):121–146

doi <https://dx.doi.org/10.15398/jlm.v11i1.315>

This work is licensed under the *Creative Commons Attribution 4.0 Public License*.

cc  <http://creativecommons.org/licenses/by/4.0/>