Title	Dissociation between Phrase Structure and Functions : A Lexical-Functional Analysis of English Negator NOT
Author(s)	Otoguro, Ryo
Citation	OUPEL(Osaka University Papers in English Linguistics). 2006, 11, p. 85-109
Version Type	VoR
URL	https://doi.org/10.18910/70805
rights	
Note	

RYO OTOGURO

DISSOCIATION BETWEEN PHRASE STRUCTURE AND FUNCTIONS: A LEXICAL-FUNCTIONAL ANALYSIS OF ENGLISH NEGATOR *NOT*[*]

1 INTRODUCTION

Negation is encoded in various ways across languages. It is one of the inflectional features in some cases, where a negative form exists in the morphological paradigm, while in other cases it is expressed purely syntactically, by negative adverbs, for instance, so that the negative meaning is added by composition at the syntactic level. Between those two types, there is also a case where negative clitics attach to the hosts. In English, negation is expressed by a negative word, *not*, which behaves like an adverb. In non-finite clauses, the parallelism is clear as shown in (1) and (2) (Baker 1995: 358):

(1) a. Jane regrets *not* having seen the movie.
    b. We asked him *not* to try to call us again.
    c. The rules require that you *not* miss the monthly meeting.
(2) a. Jane regrets *never* having seen the movie.
    b. We asked him *never* to try to call us again.
    c. The rules require that you *never* miss the monthly meeting.

The sentences in (1) are examples of gerundives, infinitival phrases, and bare verb phrases respectively, and *not* appears in front of the non-finite verbs. Those distributional properties are identical to those of a certain type of adverb such as *never*, which is shown in (2).

[*] Parts of the proposals presented in this paper originate from discussion in the Constraint-based Linguistics Discussion Group at University of Essex. I would like to thank the participants there. I am also grateful to Paul A. S. Harvey for stylistic suggestions. All remaining inadequacies are my own. This work is partly supported by a Grant-in-Aid for Young Scientists (Start-up) from the Ministry of Education, Culture, Sports, Science and Technology, Grant No. 18820019.

In finite clauses, neither *not* nor *never* can follow the verb as shown in (3). In this respect, they share a linear positional property. However, one crucial difference is that *not* requires an auxiliary element, which is often called *do*-support, whereas *never* does not as in (4):

(3) a. *I like *not* politicians.
b. *I like *never* politicians.
c. *I tear *often* the newspaper.

(4) a. *I *not* like politicians.
b. I do *not* like politicians.
c. I *never* like politicians.
d. I *often* tear the newspaper.

Another difference between *not* and adverbs in finite clauses is found in VP ellipsis. One of the constraints on VP ellipsis is prohibition of leaving an adverb behind as in (5). However, *not* is immune from this constraint, so that sentences in (6) are all acceptable (Kim 2000, Kim and Sag 2002):

(5) a. *Kim has never studied French, but Lee has *always* _____.
b. *Kim has written a novel, but Lee has *never* _____.

(6) a. Tom has written a novel, but Peter has *not* _____.
b. Kim has finished her homework, but Peter has *not* _____.

When more than one adverbial element appear in a clause, a question arises with respect to their relative scope. The well-known generalization in English is that items to the left have scope over those to the right. Hence, the following pair exhibits scopal differences (Ernst (1992: 134). See also Andrews (1983)):

(7) a. Phil has occasionally wisely refused to fight.
b. Phil has wisely occasionally refused to fight.

The interpretation of (7a) is that it is occasionally the case that Phil refuses to fight and is wise in this refusal. That is, *occasionally* has scope over *wisely*. (7b), on the other hand, means that what Phil is wise about is occasionally refusing to fight — he might have been stupid to refuse in some particular instance. Thus, it is the case where *wisely* takes wider scope than *occasionally*. A similar contrast can be found in cases where *not* appears with other adverbs:

(8) a. Mary Lou does not usually change the oil.
b. Mary Lou usually does not change the oil.

(9) a. Sam will not obviously be in trouble.
b. Sam will obviously not be in trouble.

(8a) and (9a) exemplify the cases where *not* scopes over adverbs, while (8b) and (9b) display the opposite scope relations. Thus, (8a) is interpreted as it is not the case that

Mary Lou usually changes the oil, while (8b) means that it is usual that Mary Lou does not change the oil.

Although the generalization in terms of linear ordering seems correct with respect to relative scope between adverbial elements, scope relations between auxiliaries and *not* give us a slightly different picture. In the following examples, *not* takes wider scope than the auxiliaries, even though it linearly follows them (Kim and Sag 2002: 369):

(10) a. Paul could not have worked as hard, could he?
No, he could not.
b. They will not attend the reception, will they?
c. Kim may not drink the wine on the table.

The interpretation of (10a) is that it is not the case that Paul could have worked as hard. Similarly, (10b) means that it is not the case that they will attend the reception. One interpretation of (10c) is that Kim is not permitted to drink wine on the table. Therefore, the scope generalization based on linear ordering is not valid in the case of an interaction between auxiliaries and *not*. Interestingly, however, the interpretations suggesting that auxiliaries take scope over *not* are also possible in appropriate contexts:

(11) a. Paul could [not accept the offer], couldn't he?
b. They will [not accept the offer], won't they?
c. Kim may [not drink the wine] if she doesn't like it.

(11) illustrates that the auxiliaries are not negated, as indicated by the tags. Instead, the negated verb phrases are under the scope of the auxiliaries. Hence, the interpretation of (11a) is, for instance, that it would be possible for Paul not to accept the offer. In the same vein, an appropriate interpretation of (11c) is that Kim is permitted not to drink, where *may* has scope over *not*. Thus, *not* appearing in those constructions is similar to ordinary adverbs, in that it follows the linear ordering generalization of scope relations.[1]

The difference between (10) and (11) suggests that there may be two kinds of negative constructions. The following double negation examples appear to support such an assumption (Kim and Sag (2002: 369), cf. Huddleston and Pullum (2002:

---

[1] It worth noting that a combination of some auxiliaries and *not* behaves idiosyncratically in terms of their relative scope. For instance, the combination of *must* and *not* always requires *must* to scope over *not*, while the situation is opposite in the combination of *need* and *not* (cf. Huddleston and Pullum (2002: 804)).

(i) a. They must not read it. must they?
b. They need not read it, need they?

The interpretation of (i-a) is that it is required that they not read it, while that of (i-b) is that it is not necessary for them to read it. The tag following each clause suggests that they are syntactically no different, unlike the contract found between (10) and (11).

804–5)):

(12) a. You cannot [not go with them].
b. You children may not (simply) [not do your homework] (and still pass the course).

In both examples, the auxiliaries are followed by two negators. The meanings of those sentences are: it is impossible for you not to go with them ((12a)), and children are not permitted to leave your homework undone ((12b)). Thus, those examples are thought to involve a combination of the two types of negations exemplified by (10) and (11).

The observations thus far suggest that the negator, *not*, has dual properties: it behaves like an adverb as VP modifiers in many cases, but it exhibits slightly different properties when it occurs in a finite clause. In this paper, I will present an analysis of the English negator, *not*, in one of the constraint-based frameworks, Lexical Functional Grammar (LFG) (Kaplan and Bresnan 1982, Bresnan 2001, Dalrymple 2001, Falk 2001, Kroeger 2004). My proposal is close to the analyses by Kim (2000) and Kim and Sag (2002) in another constraint-based lexicalist framework, Head-driven Phrase Structure Grammar (HPSG), in that both analyses postulate distinct structures to account for the dual properties of *not*. In Kim and Sag's HPSG accounts, adverbial properties of *not* are captured by treating it as an ordinary modifier, while they analyze *not* in finite clauses as a complement a finite auxiliary subcategorizes for. In my LFG account, the dual properties of *not* are attributed to the lexical properties of *not* itself. On the one hand, it is adjoined to a VP or other phrases as an ordinary modifier; but, on the other, it is adjoined to a finite auxiliary at the X^0 level, constituting a complex auxiliary. I will show those lexical properties of *not* provide a unified account of the problematic data introduced above.

This paper is organized as follows. In section 2, I shall introduce the basics of LFG. It starts with a discussion of surface phrase structure. I then summarize the formal and mathematical properties of functional structure as well as correspondences between surface configuration, functional structure, and the lexicon. Based on those theoretical foundations, I shall present a lexical-functional analysis of *not* in section 3. The discussion will be concluded in section 4.

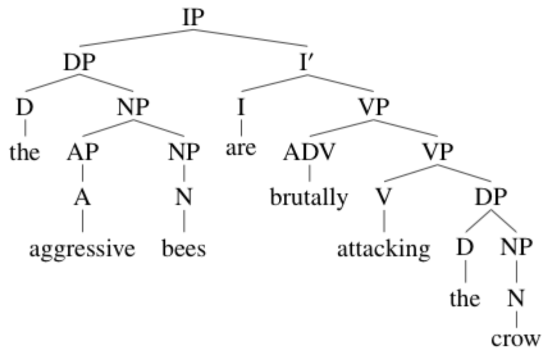
2 THEORETICAL FOUNDATIONS

LFG is a static grammar in which no derivational process like movement is implemented to produce one structure from another. Instead, LFG postulates multiple levels of representations in parallel fashion, each of which constructs different types of linguistic information such as surface phrase structure configurations, grammatical relations, and semantic relations. Those distinct structures are autonomous, in that they obey their own well-formedness conditions. The relationship between those structures is explicated by correspondences utilizing mathematical functions. Particularly relevant in this paper are two syntactic structures, c-structure (constituent-structure) and f-structure (functional-structure).

2.1 C-structure

Since Bloomfiledian grammar descriptions and the early stage of generative grammar, it has been one of the central issues in syntactic theory to investigate hierarchical and linear organizations of words. A combination of certain words behaves as a single unit, called a constituent, and that unit is further combined with other words and phrases constructing a larger constituent. The linear order between words and phrases are called precedence, while the hierarchical relationship between them is called dominance. C-structure in LFG characterizes those precedence and dominance relationships. Following the generative tradition, c-structures are represented by conventional phrase structure trees, and the permissive dominance and precedence relations are constrained by X′ schemata. For instance, the c-structure for an English sentence, *the aggressive bees are brutally attacking the crow*, can be shown as in (13):

(13)



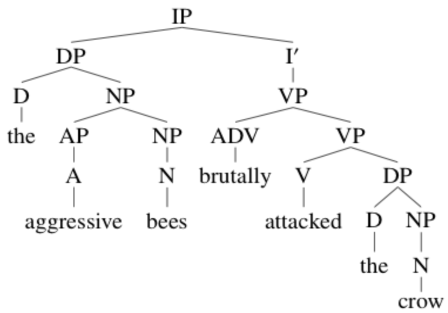
Each word is given a category label and establishes linear and hierarchical relationships with other words. The category labels such as N(oun), V(erb), A(djective), and ADV(erb) are called lexical categories. In addition to a set of lexical categories, LFG adopts functional categories such as D(eterminer), I(nflection), and C(omplementizer). The noun, *bees*, is preceded by the adjective, *aggressive*, constituting a phrase, *aggressive bees*. In this phrase, the noun is assumed to be a c-structure head, so that the whole phrase is labeled as NP. That NP is further preceded by the determiner, *the*, becoming a larger constituent, *the aggressive bees*.

In (13), the auxiliary verb, *are*, is labeled as I and heads the whole clause, IP. The idea that an auxiliary element heads a clause is originally proposed in Falk's (1984) LFG analysis of the English auxiliary system where an auxiliary is labeled as M, heading a finite clause, M‴, and it has been incorporated into transformational frameworks as well (Chomsky 1986). However, while transformational analyses assign more abstract syntactic properties to phrase structure tree, c-structure in LFG is strictly surface oriented, namely it purely represents configurations of words and phrases. This position is stated as Economy of Expression (Bresnan 2001: 91):

(14) Economy of Expression:
All syntactic phrase structure nodes are optional and are not used unless required by independent principles.

According to Economy of Expression, the sentence, *the aggressive bees brutally attacked the crow*, has the following c-structure representation:

(15)



The sentence does not have an auxiliary element that is qualified to occupy an I position. Economy of Expression prohibits the grammar to introduce such an unnecessary node in c-structure. In (15), therefore, the I′ dominates the VP without an I. In transformational analyses, the I (or T in more recent approaches) position is postulated even in this kind of situation, since functional information such as tense and agreement features are encoded in the phrase structure and it requires the existence of a phonologically null element in that position. In LFG, those abstract syntactic properties are represented in another syntactic structure called f-structure. Hence, functional categories are better regarded as referring to certain phrase structure positions, and no particular feature contents are associated with them.[2]

2.2 F-structure

As summarized above, c-structure is a representation of one kind of syntactic property,

[2] As mentioned, no derivational process is postulated in LFG, so all the phrase structure nodes are base-generated in c-structure. As a result, in some languages, finites verbs are directly base-generated in an I position. See relevant discussions in Kroeger (1993) for Tagalog, King (1995) for Russian, and Sadler (1997) and Bresnan (2001) for Welsh, for example.

i.e. precedence and dominance. However, the syntactic structure must be able to represent more than just surface constituent organizations of language. LFG postulates another level where functional relations between words and phrases are expressed. The f-structure for the sentence, *the aggressive bees brutally attacked the crow*, is shown as in (16):

(16)
$$\begin{bmatrix} \text{PRED} & \text{‘attack}\langle\text{SUBJ,OBJ}\rangle\text{’} \\ \text{SUBJ} & \begin{bmatrix}\text{PRED} & \text{‘bee’}\\ \text{NUM} & \text{PL}\\ \text{SPEC} & \text{THE}\\ \text{ADJ} & \left\{\begin{bmatrix}\text{PRED} & \text{‘aggresive’}\end{bmatrix}\right\}\end{bmatrix} \\ \text{OBJ} & \begin{bmatrix}\text{PRED} & \text{‘crow’}\\ \text{NUM} & \text{SG}\\ \text{SPEC} & \text{THE}\end{bmatrix} \\ \text{ADJ} & \left\{\begin{bmatrix}\text{PRED} & \text{‘brutally’}\end{bmatrix}\right\} \\ \text{TENSE} & \text{PAST}\end{bmatrix}$$

An f-structure is constructed by a set of ordered pairs such as ⟨NUM, SG⟩ and ⟨TENSE, PAST⟩. Those pairs are represented as attribute and value matrices as shown in (16), that is the attribute, TENSE, takes the value, PAST. Those features are atomic symbols. Atomic symbols encode syntactic properties like TENSE, ASPECT, NUM(ber), PERS(on), GEND(er), and so forth. Further, they also include GFs (grammatical functions) like SUBJ(ect), OBJ(ect), and OBL(ique) which are examples of governable GFs, and ADJ(unct), which is an example of non-governable GFs. The other type of feature appearing in f-structure are semantic forms, which are values of PRED like ‘attack⟨SUBJ,OBJ⟩’ and ‘bee’ in (16). They comprise the semantic predicate name and are sometimes followed by an argument list, namely a list of governable GFs they subcategorize for.

The governable GFs in the argument list of a semantic form must be present in the local f-structure, and conversely only the GFs specified in the argument list of a semantic form are permitted to appear in the local f-structure. Those two conditions are general constraints for f-structure, called Completeness and Coherence respectively (Kaplan and Bresnan 1982: 211–2):

(17) a. Completeness:
An f-structure is locally complete if and only if it contains all the governable grammatical functions that its predicate governs. And f-structure is complete if and only if it and all its subsidiary f-structures are locally complete.

b. Coherence:
An f-structure is *locally coherent* if and only if all the governable grammatical functions that it contains are governed by a local

> predicate. An f-structure is *coherent* if and only if it and all its subsidiary f-structures are locally coherent.

In (16), the semantic form ‘attack⟨SUBJ,OBJ⟩’ requires SUBJ and OBJ in its local f-structure. This requirement is satisfied (Completeness). The f-structure contains the only governable GFs stated in the argument list of the semantic form, i.e. SUBJ and OBJ (Coherence) — note that ADJ is not a governable GF. Thus, this f-structure turns out to be well-formed.

F-structures can also be written down by a set of propositions where a function applies to an attribute and yields a value. LFG uses the following parenthetic notation for functional application:

(18) $(f\ a) = v$ iff $\langle a\ v\rangle \in f$, where f is an f-structure, a is an atomic symbol and v is a value.

Consider a simple f-structure for a sentence, *Mary cried sadly*. If we add function names f_n as in (19), the f-structure can be described as a set of equations as in (20) — ignoring the internal structure of f_3:

(19)

$$f_1:\left[\begin{array}{ll} \text{SUBJ} & f_2:\left[\begin{array}{ll}\text{PRED} & \text{`Mary'}\\ \text{PERS} & 3\\ \text{NUM} & \text{SG}\end{array}\right]\\ \text{ADJ} & f_3:\left\{\left[\text{``sadly''}\right]\right\}\\ \text{PRED} & \text{`cry}\langle\text{SUBJ}\rangle\text{'}\\ \text{TENSE} & \text{PAST}\end{array}\right]$$

(20) $(f_1\ \text{SUBJ}) = f_2$
$(f_2\ \text{PRED}) = \text{‘Mary’}$
$(f_2\ \text{PERS}) = 3$
$(f_2\ \text{NUM}) = \text{SG}$
$(f_1\ \text{PRED}) = \text{‘cry}\langle\text{SUBJ}\rangle\text{’}$
$(f_1\ \text{TENSE}) = \text{PAST}$
$f_3 \in (f_1\ \text{ADJ})$

Following (18), the first equation in (20) states that a pair of atomic symbol, SUBJ, and f-structure, f_2, is a member of f-structure, f_1, i.e., $\langle \text{SUBJ}, f_2\rangle \in f_1$, which is true in f_1. In the same manner, $(f_2\ \text{NUM}) = \text{SG}$ and $(f_1\ \text{TENSE}) = \text{PAST}$ read as $\langle \text{NUM}, \text{SG}\rangle \in f_1$, $\langle \text{TENSE}, \text{PAST}\rangle \in f_2$, and so on. Further, since the value of ADJ is a set of f-structures, which is indicated by curly brackets, $f_3 \in (f_1\ \text{ADJ})$ holds. A set of this kind of statements is called an f-description (functional-description), and is used to specify lexical properties of words in the lexicon as we will turn to later.

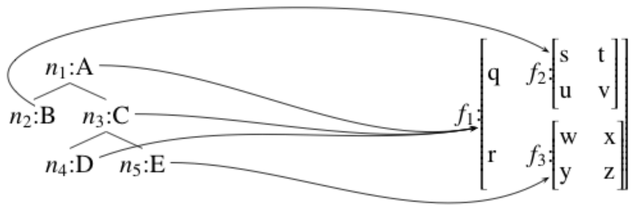
Since an f-structure is a set of ordered pairs as explained above, it can work as a mathematical function. For instance, $(f_1\ \text{TENSE}) = \text{PAST}$ is equivalent to stating that "the function f_1 is such that applying it to the argument TENSE yields the value TENSE," or f_1(TENSE) = PAST in the standard mathematical notation. As in the first equation in (20), functional application can yield another function, so a substitution like (21a) is possible, and we notate this application as in(21b):

(21) a. ((f_1 SUBJ) NUM)
= (f_2 NUM)
= SG
b. (f_1 SUBJ NUM) = SG

2.3 Correspondences

We have so far observed formal properties of two syntactic structures in LFG. They are distinct levels of representations of different linguistic features, but they are related to each other by mapping function. This function is called ϕ and maps a c-structure node, N, to an f-structure, F. Hence, the following correspondences are established:

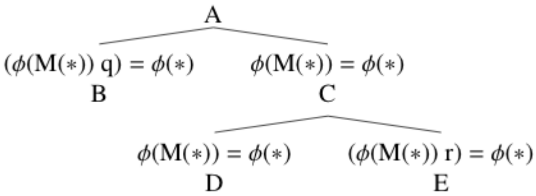
(22) $\phi: N \rightarrow F$



The arrows indicate that a c-structure node is mapped onto an f-structure. In (22), the c-structure nodes are labeled as n_1 to n_5, whereas the f-structures are named as f_1 to f_3. Thus, it shows that n_1, n_3, and n_4 are all mapped onto f_1. The complete mappings can be written as in the following equations:

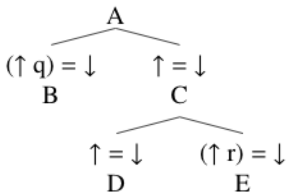
(23) $\phi(n_1) = \phi(n_3) = \phi(n_4) = f_1$
$\phi(n_2) = f_2$
$\phi(n_5) = f_3$

So far, we have described which c-structure node is mapped onto which f-structure by giving an inherent name to each c-structure node. If we assign $*$ to the current node, and refer to the mother node as M($*$) where M is a function that maps one node to its mother, the correspondences can be specified by writing functional equations onto c-structure nodes. For the c-structure in (22), the equations are assigned as in (24):

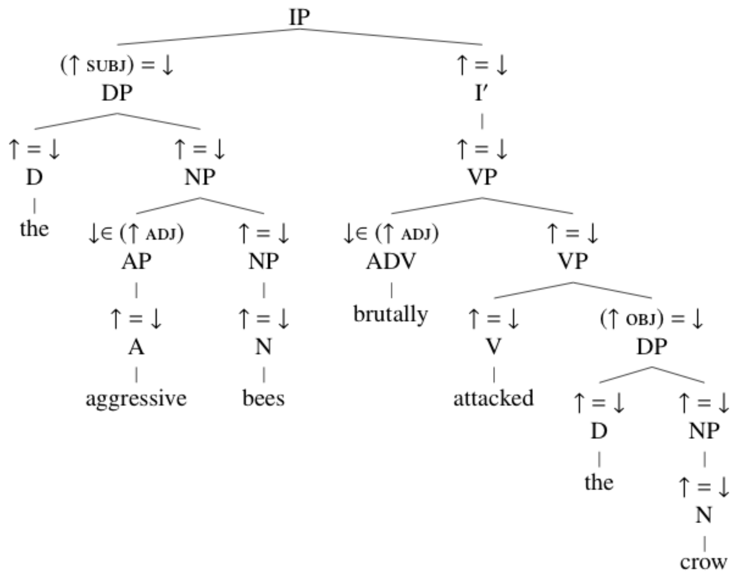
(24)


$\phi(*)$ and $\phi(M(*))$ refer to f-structures corresponding to the current node and the mother node respectively. Hence, the equation on C states that the f-structure corresponding to A node is the same as the f-structure corresponding to C node, which is true as shown in (22), namely both are mapped onto f_1. The same equation applies to D node, which in turn ensures that A, C, and D nodes all correspond to the same f-structure, f_1. The equation on B states that the f-structure corresponding to B, which is f_2, is the value of q in the f-structure, f_1, i.e., it is equivalent to $\langle q, f_2 \rangle \in f_1$ (cf. (18)). Similarly, according to the equation on E, E node is mapped onto an f-structure that is the value of r.

For expository purpose, LFG uses abbreviated notations, $\downarrow$ for $\phi(*)$ and $\uparrow$ for $\phi(M(*))$. Thus, (24) is normally written as follows:

(25)


To assign annotations to (15), the following c-structure is proposed, which has correspondences to the f-structure (16), repeated here for ease of exposition:

(26)


$$\begin{bmatrix} \text{PRED} & \text{‘attack}\langle\text{SUBJ,OBJ}\rangle\text{’} \\ \text{SUBJ} & \begin{bmatrix} \text{PRED} & \text{‘bee’} \\ \text{NUM} & \text{PL} \\ \text{SPEC} & \text{THE} \\ \text{ADJ} & \left\{\begin{bmatrix} \text{PRED} & \text{‘aggresive’} \end{bmatrix}\right\} \end{bmatrix} \\ \text{OBJ} & \begin{bmatrix} \text{PRED} & \text{‘crow’} \\ \text{NUM} & \text{SG} \\ \text{SPEC} & \text{THE} \end{bmatrix} \\ \text{ADJ} & \left\{\begin{bmatrix} \text{PRED} & \text{‘brutally’} \end{bmatrix}\right\} \\ \text{TENSE} & \text{PAST} \end{bmatrix}$$

The equations, ↑ = ↓, ensure that the IP, I′, VPs, and V are all mapped onto the same f-structure, namely the outermost f-structure. The subject DP and the D, the NPs, and the N dominated by that DP correspond to the value of SUBJ, due to (↑ SUBJ) = ↓ on the DP. Similarly, the object DP and its daughter nodes are mapped onto the value of OBJ in the f-structure. In addition, we have an adjunction as indicated by ↓ ∈ (↑ ADJ) on the AP node. It means that the f-structure corresponding to the AP node is a member of the ADJ attribute of the f-structure corresponding to its mother node, NP. The ADV receives a similar treatment.

What kind of c-structure and f-structure correspondence is assigned for a given phrase structure configuration varies across languages. But some attempts to generalize mapping principles have been made. For instance, Bresnan (2001: 102−103) proposes the following:[3]

(27) a. C-structure heads are f-structure heads.
b. Specifiers of functional categories are the grammatical discourse function DF.
c. Complements of functional categories are f-structure co-head.
d. Complements of lexical categories are the nondiscourse argument functions CF.
e. Constituents adjoined to phrasal constituents are nonargument functions AF or not annotated.

With regard to(27a), when one node is mapped onto the same f-structure as its mother, we call that node an f-structure head, i.e. that node is given ↑ = ↓. Therefore, (27a) states that if one node is a c-structure head, it corresponds to the same f-structure as

[3] Generalizing how feature contents are unified according to configurational relations is also a central issue in HPSG. In Pollard and Sag (1994), Head Feature Principle, which corresponds to (27a) in LFG, is proposed as well as a set of ID (Immediate Dominance) schemata. In more recent construction-based approaches such as Ginzburg and Sag (2000), similar generalizations are captured in terms of Generalized Head Feature Principle and nested phrase types.

its phrasal node. In (26), the D is a c-structure head of the DP, so the both nodes are mapped onto the same f-structure. The same principle applies to N and NP, A and AP, V and VP, I and IP and so on. As for (27b), a specifier of a functional category is assigned one from (↑ SUBJ) = ↓, (↑ TOPIC) = ↓, or (↑ FOCUS) = ↓, where TOPIC and FOCUS are discourse functions in f-structure. In (26), the specifier of IP is given (↑ SUBJ) = ↓, following this principle. (27c) ensures that a functional head and its complement are f-structure co-heads.[4] Hence, the NP and the VP in (26) are annotated as ↑ = ↓. (27d) is about annotating argument function to a complement of a lexical head. The object DP, a complement of the V, in (26) receives (↑ OBJ) = ↓ due to this principle. Finally, (27e) defines the annotations on adjoined phrases. In (26), the AP is adjoined to the NP, and the ADV, to the VP. They are given nonargument functions as indicated by ↓ ∈ (↑ ADJ). We will turn to more issues involving adjunction below.

2.4 The Lexicon

The correspondences between c-structure and f-structure like (26) raise a question as to where the feature contents appearing in the f-structure come from. LFG is a lexicalist theory and such features are specified in the lexical entries of words. The format of an entry is given as in (28):

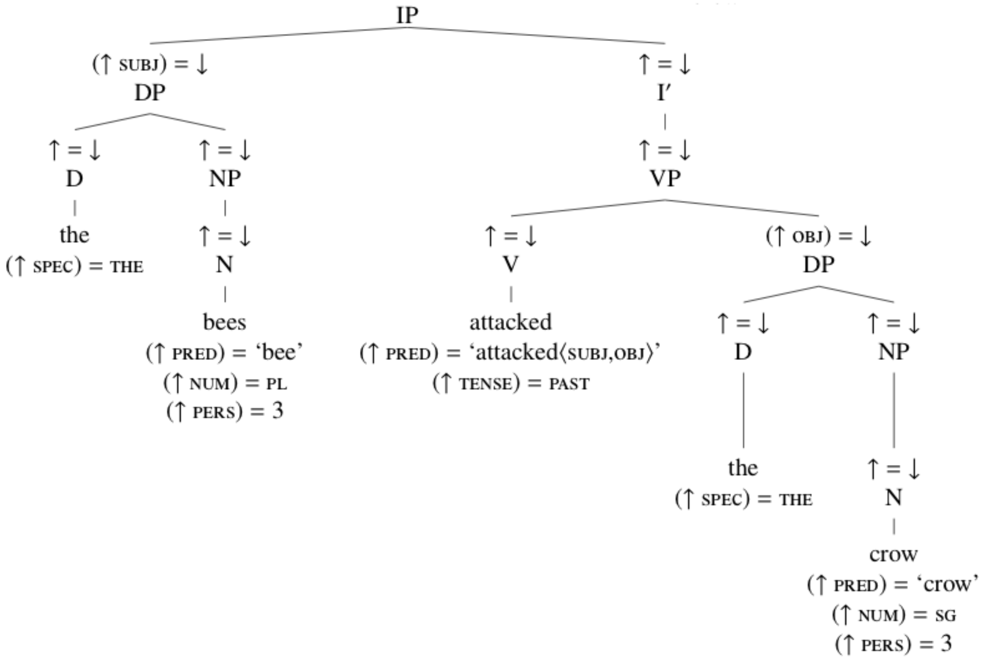
(28)	a.	*attacked*	V	(↑ PRED) = ‘attack⟨SUBJ,OBJ⟩’
				(↑ TENSE) = PAST
	b.	*bees*	N	(↑ PRED) = ‘bee’
				(↑ NUM) = PL
				(↑ PERS) = 3
	c.	*crow*	N	(↑ PRED) = ‘crow’
				(↑ NUM) = SG
				(↑ PERS) = 3
	d.	*the*	D	(↑ SPEC) = THE

The entries consist of three parts: lexical forms, category labels, and f-descriptions. Those entries are the units inserted into c-structure, so ↑ in the f-description refers to an f-structure corresponding to the pre-terminal node dominating each lexical item.

According to the feature specifications of those lexical items, a simple sentence, *The bees attacked the crow*, can be constructed in the following fashion:

[4] The idea of this f-structure co-head is close to Grimshaw’s (1991) notion of ‘Extended Projection’ where functional projections layer over a lexical category with the same categorial feature.

(29)



Following the annotations on the nodes and lexical specifications under the terminal nodes, the information flows into f-structure, and the corresponding f-structure is constructed.

3 AN ANALYSIS

3.1 VP Adjunction

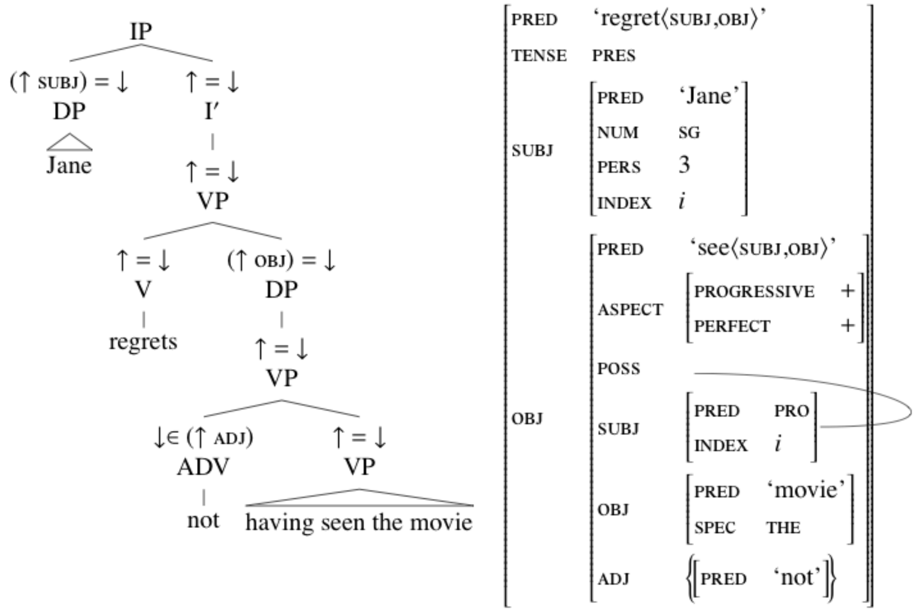
The adverbial properties of *not* summarized in section 1 can be easily captured in LFG. When *not* appear in non-finite clauses, it receives the same treatment as other adverbs such as *never*. So, the lexical entries for *not* and *never* are written as follows:

(30) a. *never* ADV (↑ PRED) = ‘never’
b. *not* ADV (↑ PRED) = ‘not’

They are given a category label, ADV, so that their distribution in phrase structure is regulated by c-structure constraints. The f-descriptions only contain PRED specifications. When they are used as a VP modifier, it is adjoined to the VP in c-structure and given the annotation, ↓ ∈ (↑ ADJ), as we have seen in the example of VP adjunction in (26).

According to the entry in (30) and the standard treatment of VP adjunction in LFG, the c-structure and the corresponding f-structure of (1) can be represented as follows:

(31) Jane regrets not having seen the movie.



(31) involves a gerundive complement, which makes the structures slightly complicated. Since the analysis of gerunds is not a central issue in this paper, I simply follow the existing proposal.[5] The gerundive VP is dominated by a DP that is annotated as (↑ OBJ) = ↓, so that it is mapped onto OBJ in the f-structure. The value of OBJ in the f-structure consists of a SUBJ whose PRED value is PRO. This SUBJ is anaphorically controlled by the matrix SUBJ corresponding to *Jane* by INDEX *i*. Since a subject of a gerundive predicate is marked by possessive *'s* when it is overtly expressed, the f-structure involves POSS. In (31), however, it is PRO that functions as a subject of the gerund, so the POSS and the SUBJ are identified by functional control as notated by the connecting line. Crucial in those structures is the ADV, *not*. It is adjoined to the VP node and mapped onto the value of ADJ contributing PRED value. This analysis is the same as ordinary adverbs, so replacing *not* with *never* simply end up with having a different PRED value in the ADJ.[6]

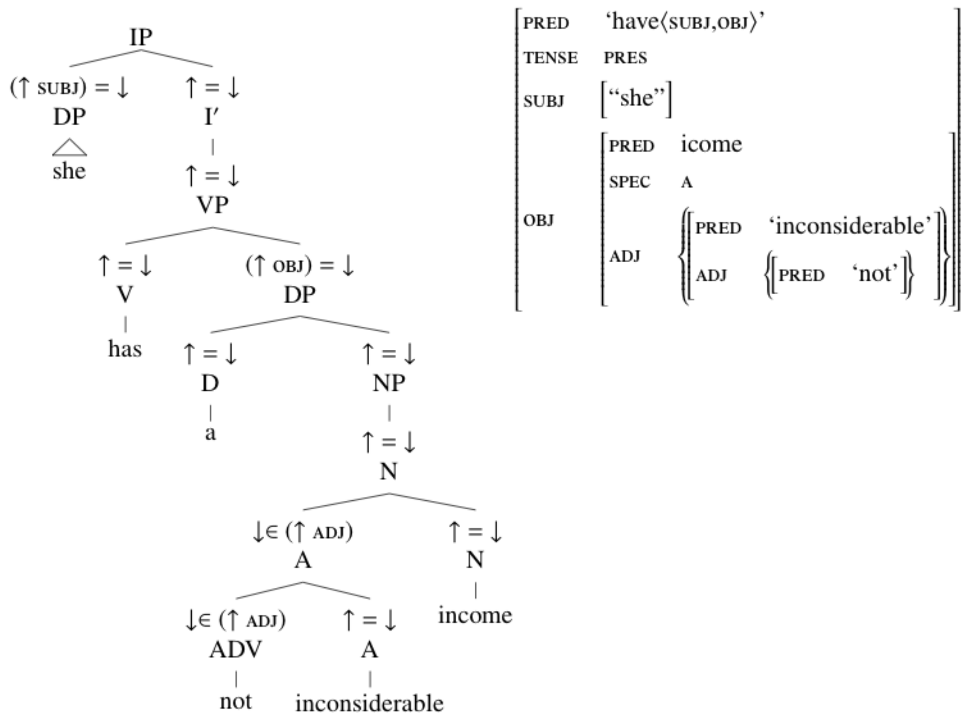
---

[5] I refer readers to Bresnan (2001: 287–301) for details of the proposal summarized here.

[6] In LFG, negative elements are normally thought to contribute ⟨NEG, +⟩ feature to f-structure. However, Sells (2001) also proposes to treat negation is a type of adjunct. I will discuss this point below in the analysis of another type of *not*.

The adverbial use of *not* modifies other categories as well as VP. For instance, it can negate a prenominal adjective as in *She has a not inconsiderable income* (Huddleston and Pullum 2002: 788). This sentence can be analyzed as in (32):

(32) She has a not inconsiderable income.



Following Sadler and Arnold (1994), I treat a prenominal adjective modifier as an X^0 adjunct, constituting A-N ‘small’ constructions. Although adjunction to X^0 level deviates from the standard X′ theory, it has been utilized in a number of LFG analyses for some languages (Sells 1994, 1998, 2001, Sadler 1997, Toivonen 2001). The whole prenominal modifier, *not inconsiderable*, is mapped onto the value of ADJ in the OBJ. The adverbial *not* corresponds to another ADJ inside the f-structure corresponding to the whole prenominal modifier so that it negates the adjective, *inconsiderable*.

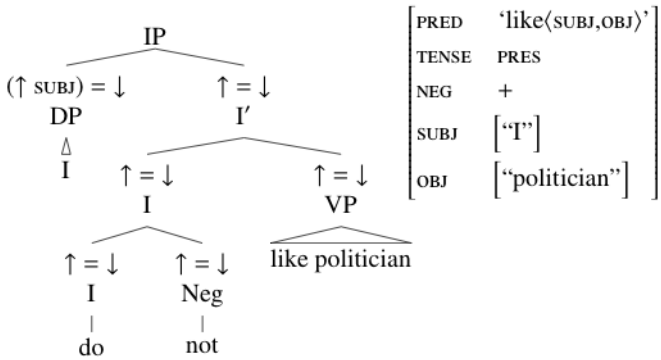
3.2 I^0 Adjunction

The treatment of *not* introduced above captures its similarity to adverbs. However, it leaves some peculiar features unexplained. As we have seen in section 1, *not* always requires an auxiliary in finite clauses; it can be stranded under VP ellipsis; and finally, it can take wider scope over a preceding auxiliary. To account for these properties, I shall argue that *not* has another lexical entry as follows:

(33) *not* Neg $(\uparrow \text{NEG}) = +$
$(\mu(\text{M}(*))\ \text{AUX}) =_c +$
$(\mu(\text{M}(*))\ \text{FIN}) =_c +$

One notable difference from the adverbial *not* is the category. I assume that Neg is right-adjoined to I^0 node, so that the combination of a finite auxiliary and *not* constitute a complex I. The first equation states that it contributes ⟨NEG, +⟩ to the clause. The second and third equations will be explained below. By having this entry, a finite clause with *not* has the following structure:

(34) I do not like politicians.



The Neg adjoined to the finite auxiliary is annotated as $\uparrow = \downarrow$, so it is an f-structure co-head with the auxiliary, *do*, and the lexical head of the complement VP. As a result, sentential negation is encoded in the f-structure by ⟨NEG, +⟩ feature.[7]

One of the important constraints on *not* in finite clauses is that it always requires a finite auxiliary. This constraint is formalized in the lexical entry (33). It involves the function μ. In section 2, we have introduced ϕ that maps a c-structure node to an f-structure. In addition to c-structure and f-structure, recent works in LFG postulate other structures to represent various linguistic properties, which is called ‘projection architecture.’ In that architecture, multiple levels of representations have correspondences, and μ is proposed to link c-structure to m-structure (morphological/ morphosyntactic structure) (Butt et al. 1996) or f-structure to m-structure (Frank and Zaenen 2002, Otoguro 2006, 2007). In (33), I take the former correspondences. Recall that M is a function that maps a c-structure node to its mother, so $\mu(\text{M}(*))$ refers to an m-structure corresponding to its mother node, Neg. $=_c$ is called a constraining equation that requires the presence of an attribute-value pair specified by the equation. Within the projection architecture, the lexical entries contain specifications of

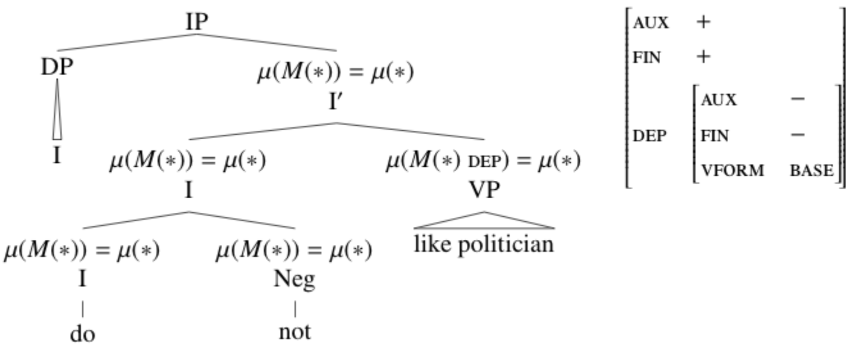
[7] The category label, Neg, is not meant to restrict I^0 adjunction to *not*. There are some other candidates to occupy this position such as *so* and *too* (cf. Kim and Sag 2002: 374–375).

properties not only for f-structure properties, but also for other structures. Thus, the auxiliary, *do*, and the verb, *like*, appearing in (34) have the following descriptions:

(35) a. *do* I ($\uparrow$ TENSE) = PRES
(μ(M($*$)) AUX) = +
(μ(M($*$)) FIN) = +
(μ(M($*$)) DEP FIN) $=_c$ –
(μ(M($*$)) DEP VFORM) $=_c$ BASE
b. *like* V ($\uparrow$ PRED) = 'like⟨SUBJ,OBJ⟩'
(μ(M($*$)) AUX) = –
(μ(M($*$)) FIN) = –
(μ(M($*$)) VFORM) = BASE

AUX is a feature to state whether the entry is an auxiliary or not, so it is set positive for *do* and negative for *like*. FIN controls whether the verb is finite (+) or nonfinite (–). DEP represents a morphosyntactic dependency. The concept of morphosyntactic dependency would be clearer when we consider the combination of *have* and a past participle in the perfect and *be* and a present participle in the progressive. The morphological forms of lexical verbs in those combinations are constrained by the auxiliary verbs, *have* and *be*. A similar asymmetric dependency relation is observed between *do* and *like*. The reason that *like* is in the base form is that it is dependent on the auxiliary, *do*. Those morphological constraints are formalized in the constraining equations in (35). (μ(M($*$)) DEP FIN) $=_c$ – and (μ(M($*$)) DEP VFORM) $=_c$ BASE state that *do* requires its morphosyntactic dependent to contribute ⟨FIN, –⟩ and ⟨VFORM, BASE⟩ to the values of DEP. Those constraints are satisfied by the last two equations in the entry (35b) . Following those m-equations, the annotated c-structure for μ-projection and the m-structure corresponding to the I′ node will be as follows:

(36)

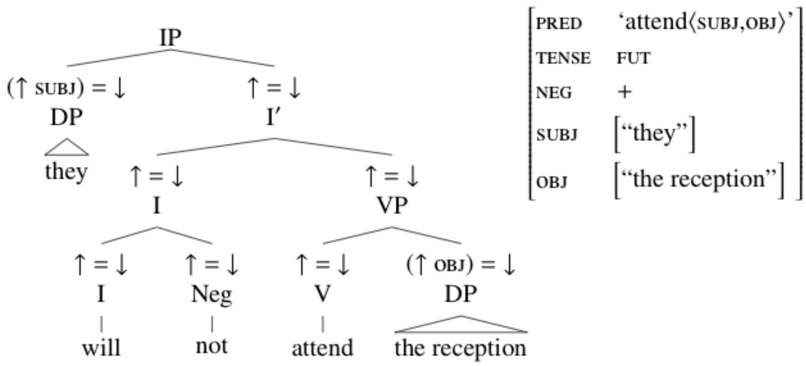


Turning back to the constraining equations in (33), (μ(M($*$)) AUX) $=_c$ + and (μ(M($*$)) FIN) $=_c$ + means that *not* requires the pair ⟨AUX, +⟩ and ⟨FIN, +⟩ to be present in the m-structure corresponding to its mother. Those features are provided by a finite auxiliary to the m-structure, so that the constraining equations in (33) essentially require the existence of a finite auxiliary. In the case of *I do not like politicians*, those requirements are satisfied as illustrated in (34) and (36).

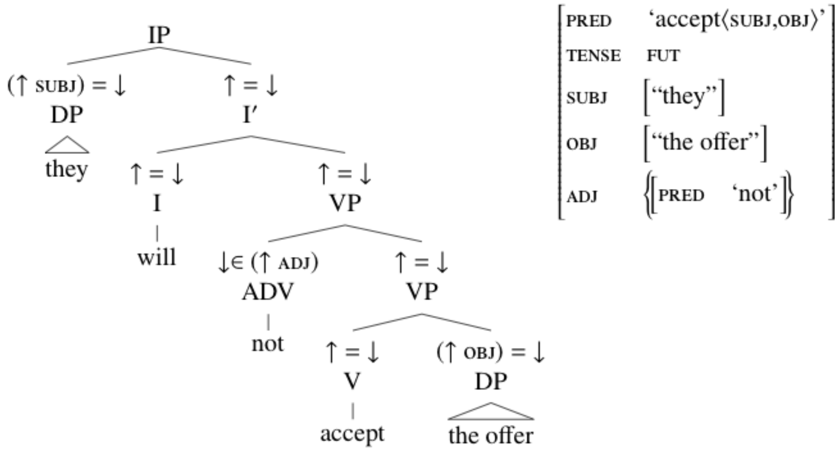
3.3 Featural Differences

It is worth noting that *not* in (33) differs from the adverbial *not* introduced in (30) not simply in the phrase structure configurations, i.e., I^0 adjunction vs. VP adjunction (cf. (32) in which the adverbial *not* is adjoined to A^0). The crucial difference is that the former introduces ⟨NEG, +⟩ to the clause, while the latter functions exactly like an adverb being mapped onto the value of ADJ in the f-structure. In the terminology often used in the literature, the negation by ⟨NEG, +⟩ can be identified as sentential or clausal negation, whereas the negation by ADJ, as constituent negation. We have already observed the distinct behaviors between the two types in section 1. The contrast between (10) and (11), for instance, highlights the structural and functional differences. When *not* appears between a finite auxiliary and a non-finite verb, it is ambiguous between the I^0-adjoined *not* and the VP-adjoined adverb *not*. My proposal would predict structures like (37) for the former case and (38) for the latter:

(37)



(38)



What kind of structure tag questions like (10) and (11) take has not been deeply investigated in LFG, so I leave the question open. But it is certain that the tag must be able to access the polarity of the main clause, so that it can formally express the opposite polarity. Considering the fact that the tags in (10) are in the affirmative, while those in (11) are in the negative, the main clauses of those two types must have different values for polarity. (37) and (38) express such a contrast in the f-structures. (37) has ⟨NEG, +⟩ in the f-structure corresponding to the whole clause, so that the tag is able to detect this feature and set its own polarity positive. In (38), on the other hand, despite the existence of *not* in the c-structure, it does not contribute ⟨NEG, +⟩, namely the whole clause is not negative in terms of f-structure feature. As a result, the tag attached to this clause can access that affirmativeness and set its polarity negative. If we assume that all the instances of *not* introduce ⟨NEG, +⟩, this kind of contract would not be captured.[8] If all the examples of *not* function as ADJ, the contrast would still be unexplained. Hence, the empirical data support the analysis having two types of *not*: one for sentential negation, and the other for constituent negation.

A further support for associating ⟨NEG, +⟩ with *not* adjoined to I^0, but not with the adverbial *not*, comes from the existence of the inflectional negative auxiliary forms. As argued by Zwicky and Pullum (1983), English contracted auxiliaries such as *won't*, *doesn't*, and *haven't* are actually inflected forms, that is they inflect for a negative feature. Interestingly, those inflectional negative auxiliaries always function as sentential negation as in (39):[9]

---

[8] Butt et al. (1999: 137) point out the problem of identical f-structures between clausal negation and VP negation by introducing ⟨NEG, +⟩ in both cases. The argument here reveals that the problem is more than just scope of negation.

[9] Huddleston and Pullum (2002: 801) gives the following exception:

(i) a. He often isn't there when you call him, isn't he?
 b. He often isn't there when you call him, so is his secretary.

The position of *often* indicates that *isn't* may not be in the I position, and further *is* in this use is a verb of existence. Thus, I regard this is an exceptional case where a contracted negative form appearing in the V position, and function as constituent negation. Also note that those sentences are sharply

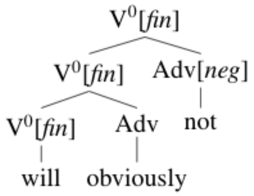
(39) a. They couldn't accept the offer, could they/*couldn't they?
b. They won't accept the offer, will/*won't they?

In (39), the auxiliaries in the main clauses are inflectional negative forms, and they always require positive tags. Hence, they introduce sentential negation, rather than VP constituent negation. Considering my proposal that an auxiliary and *not* constitute a complex I functioning as sentential negation, it would not be surprising that those inflectional negative forms are developed from the I^0 adjunction structures, and the negative feature enters the morphological inflectional system.[10]

3.4 Phrase Structure and VP Ellipsis

Kim and Sag (2002: 363) argue against the proposal that *not* adjoined to V^0, for the reason that the analysis would be forced to assumed that both *obviously* and *not* are adjoined to V^0, *will*, in a sentence like *They will obviously not have time to change*, as in (40). In my analysis, however, this is an example of the adverbial *not*. Thus, the structure would be shown as in (41):

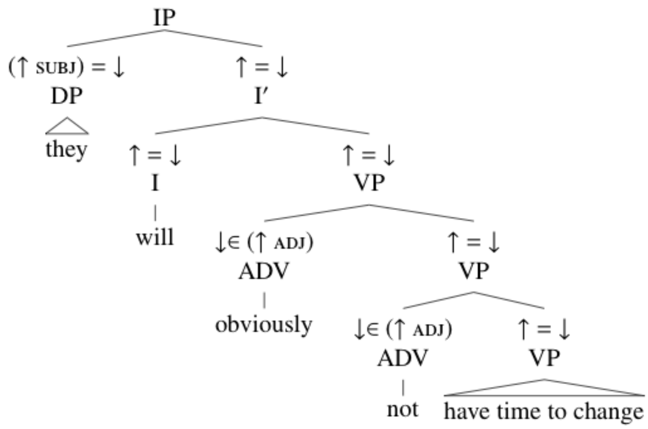
(40)



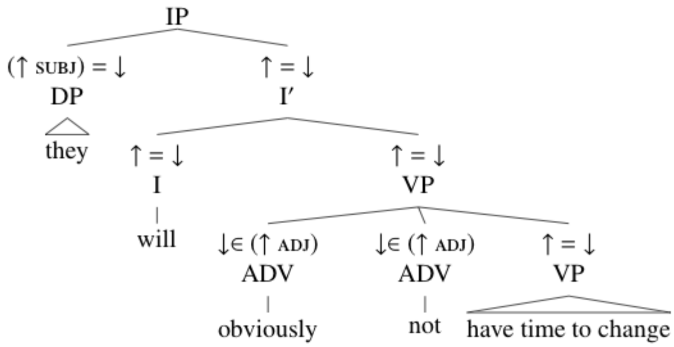
ungrammatical to some speakers.

[10] Of course, it is shortcoming to argue that the availability of a certain feature in the inflectional system ensures the existence of the same feature in the syntax. But it is not uncommon that seemingly syntactic properties become a part of morphological feature system, pronouns become pronominal affixes in many languages, for example.

(41) a.



b.



As obvious from the discussion of c-structure, LFG does not restrict phrase structure tree to binary branching. Hence, multiple adjunctions to a single VP like (41b) are possible. Since both c-structures in (41) construct exactly the same f-structure, the choice between them must be made as to which VP behaves as a constituent. As far as I am aware, there is no evidence for *not have time to change* to constitute a single unit (cf. Dalrymple (2001: 48–49)), so I assume that (41) is the correct structure.

It might be claimed that structures like (41) would better capture the relative scope between the two adverbs, as seen in (8) and (9). However, since LFG's c-structure is a representation of surface 'constituents,' properties like scope differences do not straightforwardly justify the layered structures. As argued in Dalrymple (2001: 49), recent work on semantics in LFG provides a way to capture relative scope differences in terms of meaning assembly by Glue logic, not by phrasal configuration. Since introducing Glue Semantics in LFG is beyond the scope of this paper, here I simply state that the relative scope among ADJ modifiers is captured in a different level, semantic structure. With respect to the relative scope between a finite auxiliary and *not* shown in (10), the situation is different. In my proposal, *not* occurring with a finite auxiliary is not an adverb, but the I^0-adjoined negator. Since it introduces ⟨NEG, +⟩ to the f-structure of a whole clause, it is a natural consequence to have scope over the finite auxiliary, though there are some idiosyncratic cases as mentioned in footnote 1.

Finally, I will outline how VP ellipsis can be accounted for in the proposal. We have observed that VP ellipsis is not applicable to the elements following an adverb as in (5) (repeated here as (42)). However, a VP following *not* can undergo ellipsis as shown in (6) (repeated here as (43)):

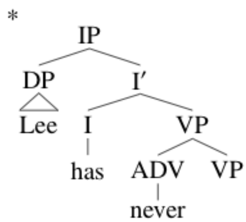
(42) a. *Kim has never studied French, but Lee has *always* _____.
b. *Kim has written a novel, but Lee has *never* _____
(43) a. Tom has written a novel, but Peter has *not* _____.
b. Kim has finished her homework, but Peter has *not* _____.

However, not all the instances of *not*, allow ellipsis of the following phrases. If it does not immediately follow a finite auxiliary, ellipsis is not allowed as in (44) (Kim 2000: 134):

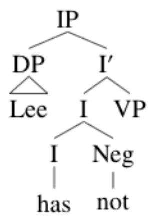
(44) a. *Susan may have been studying but Mary may have been not _____.
b. *Susan may have been studying but Mary may have not _____.

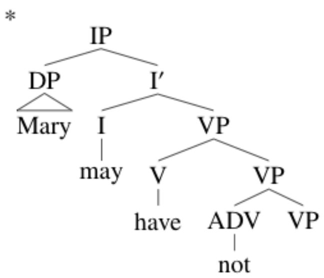
Thus far, the generalization seems that ellipsis is only applicable to a VP complement of a finite auxiliary as indicated by the following c-structures:

(45) a.



b.





However, careful observation reveals that that is an incorrect generalization. The following examples illustrate that a complement of a non-finite verb can also undergo ellipsis (Kim and Sag 2002: 368):

(46) a. Lee may have been studying too much recently, but I think that Kim may not have _____.

b. Lee may have been studying too much recently, but I think that Kim may have not been _____.

Instead, the phrases to which ellipsis is applicable is a complement of an auxiliary regardless of its finiteness. The reason that (42) and (43) are unacceptable is that the adverbial modifiers lack their host VP, i.e., there is nothing for them to modify. On the other hand, *not* adjoined to I^0 is treated as a part of the finite auxiliary, since it constitutes a complex I with the auxiliary. Therefore, ellipsis can apply to the complement of a complex I.

Formalization of the target of VP ellipsis is attained by referring to the c-structure node, which is a value of DEP in m-structure — recall that a complement of an auxiliary is a morphosyntactic dependent and mapped onto the value of DEP. Hence, I propose the following generalization of VP ellipsis:

(47) VP Ellipsis Generalization:
Allow the c-structure node μ^{-1}(DEP) to be ellipted.

Since μ is a function that maps a c-structure node to an m-structure, the inverse function, μ^{-1}, operates in the opposite way, i.e., it maps an m-structure to c-structure nodes. In (47), the inverse function takes DEP as its argument, so it returns a set of c-structure nodes that are mapped onto the value of DEP in m-structure. Since such nodes are complements of an auxiliary, (47) essentially selects the target to which VP ellipsis is allowed.

4 CONCLUSION

In this paper, I have illustrated how the puzzling properties of *not* can be resolved in the constraint-based lexicalist framework, LFG. As summarized in section 1, *not* exhibits adverbial properties, on the one hand, but it also shows distinct behavior in finite clauses, with respect to relative scope to a finite auxiliary, VP ellipsis, and its requirement of a finite auxiliary exemplified by *do*-support. To account for those peculiar features of *not*, I have proposed two separate lexical entries. The one is similar to ordinary adverbs, which modifies VP, A and other words and phrases. Under an LFG analysis, it functions as ADJ in f-structure. This lexical entry explains why *not* behaves like adverbs in non-finite clauses, and sometimes in finite clauses as well.

Crucially, I also proposed another lexical entry for *not*. It is given a distinct category label, Neg. Incorporating recent insights in the LFG literature, I propose that Neg is adjoined to I^0 level node. The combination of a finite auxiliary and *not* function as I altogether, and *not* contribute ⟨NEG, +⟩ to the clause, which explains clausal negation properties of *not*, contrasting to constituent negation by the adverbial *not*. Further, presence and absence of the negative feature can neatly account for the change of polarity of tag under tag questions. The requirement of a finite auxiliary is also captured by the proposal, utilizing the projection architecture in LFG. I argue that m-structure project off c-structure, and represent morphosyntactic dependency relation between auxiliaries. By referring to the AUX and FIN features in m-structure, the constraint of presence of a finite auxiliary is formalized. Moreover, m-structure plays a crucial role in accounting for VP ellipsis. A careful observation reveals that VP ellipsis is applied to a complement of an auxiliary, regardless of its finiteness. This generalization is stated by the inverse function of μ-projection. In sum, this paper shows that a lexicalist approach gives a straightforward account of the seemingly complex phenomena, and the coverage of the analysis suggests a promising way of investigating linguistic data in terms of interaction between lexical features, surface configuration, functional relation, and morphosyntactic properties.

REFERENCES

Andrews, Avery D. III (1983) "A Note on the Constituent Structure of Modifiers," *Linguistic Inquiry* 14, 695–697.

Baker, Carl. L. (1995) *English Syntax*, Second Edition, MIT Press, Cambridge, Mass.

Bresnan, Joan (2001) *Lexical-Functional Syntax*, Blackwell, Oxford.

Butt, Miriam, María-Eugenia Niño, and Frédérique Segond (1996) "Multilingual Processing of Auxiliaries within LFG," in Dafydd Gibbon (ed.) *Natural Language Processing and Speech Technology*, 111–122, Mouton de Gruyter, Berlin, reprinted in Louisa Sadler and Andrew Spencer (eds.) *Projecting Morphology*, 11–22, CSLI Publications, Stanford.

Butt, Miriam, Tracy Holloway King, María-Eugenia Niño, and Frédérique Segond (1999) *A Grammar Writer's Cookbook*, CSLI Publications, Stanford.

Chomsky, Noam (1986) *Barriers*, MIT Press, Cambridge, Mass.

Dalrymple, Mary (2001) *Syntax and Semantics 34: Lexical Functional Grammar*, Academic Press, San Diego.

Ernst, Thomas (1992) "The Phrase Structure of English Negation," *The Linguistic Review* 9, 109–144.

Falk, Yehuda N. (1984) "The English Auxiliary System: A Lexical-Functional Analysis," *Language* 60, 483–509.

Falk, Yehuda N. (2001) *Lexical-Functional Grammar: An Introduction to Parallel Constraint Based Syntax*, CSLI Publications, Stanford.

Frank, Anette, and Annie Zaenen (2002) "Tense in LFG: Syntax and Morphology," in Hans Kamp and Uwe Reyle (eds.) *How We Say WHEN It Happens: Contributions to the Theory of Temporal Reference in Natural Language*, 17–52, Max Niemeyer Verlag, Tübingen, reprinted in Louisa Sadler and Andrew Spencer (eds.) *Projecting Morphology*, 23–65, CSLI Publications, Stanford.

Ginzburg, Jonathan, and Ivan A. Sag (2000) *Interrogative Investigations: The Form, Meaning, and Use of English Interrogatives*, CSLI Publications, Stanford.

Grimshaw, Jane (1991) "Extended Projection," ms. Rutgers University.

Huddleston, Rodney, and Geoffrey K. Pullum (2002) *The Cambridge Grammar of the English Language*, Cambridge University Press, Cambridge.

King, Tracy Holloway (1995) *Configuring Topic and Focus in Russian*, CSLI Publications, Stanford.

Kroeger, Paul (1993) *Phrase Structure and Grammatical Relations in Tagalog*, CSLI Publications, Stanford.

Kroeger, Paul (2004) *Analyzing Syntax*, Cambridge University Press, Cambridge.

Kaplan, Ronald M. (1995) "The Formal Architecture of Lexical-Functional Grammar," in Mary Dalrymple, Ronald M. Kaplan, John T. Maxwell III, and Annie Zaenen (eds.) *Formal Issues in Lexical-Functional Grammar*, 7–27, CSLI Publications, Stanford.

Kaplan, Ronald M., and Joan Bresnan (1982) "Lexical-Functional Grammar: A Formal System for Grammatical Representation," in Joan Bresnan (ed.) *The Mental Representation of Grammatical Relations*, 173–281, MIT Press, Cambridge, Mass.

Kim, Jong-Bok (2000) *The Grammar of Negation: A Constraint-Based Approach*, CSLI Publications, Stanford.

Kim, Jong-Bok, and Ivan A. Sag (2002) "Negation Without Head-Movement," *Natural Language and Linguistic Theory* 20, 339–412.

Otoguro, Ryo (2006) *Morphosyntax of Case: A Theoretical Investigation of the Concept*, Doctoral Dissertation, University of Essex.

Otoguro, Ryo (2007) "Formal Aspects of Morphosyntax in LFG," *English Linguistics* 24, 256–278.

Pollard, Carl, and Ivan A. Sag (1994) *Head-Driven Phrase Structure Grammar*, University of Chicago Press and CSLI Publications, Chicago and Stanford.

Sadler, Louisa (1997) "Clitics and the Structure-Function Mapping," in Miriam Butt and Tracy Holloway King (eds.) *Proceedings of the LFG97 Conference*, CSLI Publications, Stanford.

Sadler, Louisa, and Douglas J. Arnold (1994) "Prenominal Adjectives and the Phrasal/Lexical Distinction," *Journal of Linguistics* 30, 187–226.

Sells, Peter (1994) "Subphrasal Syntax in Korean," *Language Research* 30, 351–386.

Sells, Peter (1998) "Scandinavian Clause Structure and Object Shift," in Miriam Butt and Tracy Holloway King (eds.) *Proceedings of the LFG98 Conference*, CSLI Publications, Stanford.

Sells, Peter (2001) *Structure, Alignment and Optimality in Swedish*, CSLI Publications, Stanford.

Toivonen, Ida (2003) *Non-Projecting Words: A Case Study of Swedish Particles*, Kluwer Academic Publishers, Dordrecht.

Zwicky, Arnold M., and Geoffery K. Pullum (1983) "Cliticization vs. Inflection: English *n't*," *Language* 59, 502–513.

Ryo Otoguro
otoguro@fpu.ac.jp