# An inquisitive account of *wh*-questions through event semantics

Maxime Amblard, Maria Boritchev, and Philippe de Groote

Université de Lorraine, CNRS, Inria, LORIA, F-54000 Nancy, France
maxime.amblard@loria.fr, maria.boritchev@loria.fr,
philippe.degroote@loria.fr

**Abstract.** We give a compositional account of the semantics of *wh*-questions. This account is based on the alliance of two semantic theories: neo-Davidsonian event semantics [22], on the one hand, and inquisitive semantics [6], on the other hand. The resulting system is implemented in the framework of the abstract categorial grammars [13].

## 1 Introduction

Modeling the meaning of sentences in natural languages is a task that can be approached from different perspectives, ranging from distributional semantics to formal semantics. The study presented in this paper is conducted from the point of view of and in the scope of formal semantics.

There are multiple formal accounts of the semantics of declarative sentences, which mainly derive from Montague's seminal work [18–20]. Most of these accounts are truth-conditional: they characterize the semantics of a declarative sentence by specifying the conditions under which that declarative sentence can be considered as true. In the present work, our object of study is interrogative sentences, and for this type of sentence, truth-conditional approaches do not apply immediately. It is indeed not clear what assigning a truth value to a question would mean [12]. Is a yes/no-question true or false? What is a negative answer to a yes/no-question? What does it mean for a *wh*-question to be true or false?

Soon enough following Montague's work, Hamblin proposed a solution to this problem [16]. His proposal consists in characterizing the semantic content of a question by specifying its set of possible answers. This gives rise to a semantic framework known as *alternative semantics*. The more recent theory of Inquisitive semantics [6], which belongs to this tradition, makes several technical improvements over other Hamblin-like theories. It provides a logic that handles interrogative and declarative sentences without differentiation, and that is amenable to a compositional treatment, as shown in [7].

Following classic literature on formal semantics of interrogative sentences (see [11, 24], for a survey), we investigate questions through the types of the queries they raise. This leads us to envision an approach based on a semantic framework that makes extensive use of thematic roles: neo-Davidsonian event semantics [22]. Then, the main idea behind our proposal is to see, at the semantic level, a *wh*-extraction as an inquisitive existential quantification. This quantification binds the variable that serves as an argument to the thematic role corresponding to the *wh*-word that triggers the *wh*-extraction.

The paper is organized as follows. The next section presents some useful linguistic and mathematical preliminaries. In particular, it encompasses a discussion about the notion of thematic role, and brief introductions to neo-Davidsonian semantics, inquisitive logic, and abstract categorial grammars. Section 3 is the core of the paper, in which we formally present our proposal. We parallel *wh*-extraction with quantifier raising, which brings us to consider *wh*-words as generalized quantifiers. The resulting system is then formalized in terms of abstract categorial grammars. These grammars are kept as simple as possible with the aim of capturing the gist of the semantic interpretation of *wh*-questions. This simplicity, however, is not without its drawbacks. It is indeed the case that the grammatical system defined in Section 3 overgenerates and introduces spurious ambiguities. In order to remedy this, we define in section 4 a device that allows us to control overgeneration. This device takes the form of an additional abstract categorial grammar that we add to our grammatical architecture. Finally, in Section 5, we conclude.

## 2 Linguistic and mathematical preliminaries

### 2.1 *Wh*-questions and thematic roles

As defined in [10], *wh*-questions, in English, are questions that give rise to answers whose semantic sorts match those of the *wh*-phrase contained in the interrogative. A *wh*-phrase is introduced by a *wh*-word: `what, when, where, who, whom, which, whose, why, how` [1].

For this definition to be operational, it is necessary to systematically define the semantic sorts. A way to do so is by using *thematic roles*, as inspired by [8, 17]. This raises many discussions related to the interpretation and definition of thematic roles. To tighten up, we use the following list of thematic roles, which is inspired by Fillmore's and Gruber's works [9, 15]: `participant`, `actor`, `cause`, `agent`, `undergoer`, `instrument`, `theme`, `pivot`, `patient`, `attribute`, `location` (see the definitions in Table 1). This list is in fact adapted from [3], without thematic roles specific to events with symmetrical participants, events of perception, or events of communication, and with the addition of the `Location` role from [4].

We note that a more detailed list of thematic roles is presented in the DIT++ schema [4], a semantically based framework for the analysis and annotation of dialogue. Following the statement in [2] that no fixed list of thematic roles can be established (nor crosslinguistic, nor for English only), we choose to showcase our method on a shorter list for the sake of readability. Our model can be easily tailored to a different list of thematic roles.

Once the thematic roles are set, asking a *wh*-question corresponds to interrogating the content of a thematic role. Therefore, in order to express the semantics of *wh*-questions, we need a formalism that gives an explicit access to terms or

**Table 1.** List of thematic roles

| **Thematic role** | **Definition** |
|---|---|
| Participant | Entity involved in a state or event |
| Actor | Participant that is the instigator of an event |
| Cause | Actor (animate or inanimate) in an event, that initiates the event, without intentionality or consciousness, existing independently of the event |
| Agent | Actor in an event who initiates and carries out the event intentionally or consciously, existing independently of the event |
| Undergoer | Participant in a state or event that is not an instigator of the event or state |
| Instrument | Undergoer that is central to an event or state that is not an instigator of the event or state |
| Theme | Undergoer that is central to an event or state that does not have control over the way the event occurs, is not structurally changed by the event, and/or is characterized as being in a certain position or condition throughout the state |
| Pivot | Theme that participates in an event with another theme unequally but is central to the event |
| Patient | Undergoer in an event that experiences a change of state, location, or condition, that is causally involved or directly affected by other participants, and exists independently of the event |
| Attribute | Undergoer that is a property of an entity or entities, as opposed to the entity itself |
| Location | Place where an event occurs or a state is true |

variables corresponding to the content of thematic roles. neo-Davidsonian event semantics [22] is such a formalism.

### 2.2 Neo-Davidsonian event semantics

Neo-Davidsonian event semantics (NDES) is a formalism in which every sentence is considered in terms of occurring events and ways the sentence semantic constituents relate to this event. Recent updates such as [5, 23] present compositional versions of NDES.

Neo-Davidsonian semantics can be formalized using a simple type theory with three atomic types:

$$\begin{array}{rl} \mathbf{e} & \text{(entities)} \\ \mathbf{t} & \text{(truth values)} \\ \mathbf{v} & \text{(events)} \end{array}$$

$\mathbf{e}$ and $\mathbf{t}$ are inherited from Montague [20], while $\mathbf{v}$ is introduced as the type of events. To illustrate the approach, let us give a simple example taken from [5]. Consider the following sentence:

(1) John kissed Mary

A classical Montague grammar would express the semantics of (1) by a simple atomic formula akin to the following one:

(2) $\mathbf{kiss}\,\mathbf{john}\,\mathbf{mary}$

where $\mathbf{kiss}$ is of type $\mathbf{e} \to \mathbf{e} \to \mathbf{t}$, and $\mathbf{john}$ and $\mathbf{mary}$ are of type $\mathbf{e}$.

By contrast, a Montague grammar based on NDES would interpret (1) as follows:

(3) $\exists e.\, (\mathbf{kiss}\, e) \wedge (\mathbf{agent}\, e\, \mathbf{john}) \wedge (\mathbf{patient}\, e\, \mathbf{mary})$

whith $\mathbf{kiss}$ of type $\mathbf{v} \to \mathbf{t}$, and $\mathbf{agent}$ and $\mathbf{patient}$ of type $\mathbf{v} \to \mathbf{e} \to \mathbf{t}$. The intuition behind this neo-Davidsonian interpretation may be grasped by paraphrasing (3) as follows: *there is a kissing event the agent of which is John and the patient of which is Mary.*

One of the main arguments in support of NDES is its flexibility with regard to the treatment of the optional arguments of the verbs. To exemplify it, let us add to Example (1) an adverbial modifier.

(4) John kissed Mary in the garden

In Montague semantics, a verb phrase is interpreted as (the intension of) a set of entities. Accordingly, an adverbial modifier is interpreted as a set transformer. With such an approach, the semantic interpretation of (4) might be as below:

(5) $\mathbf{in}\,\mathbf{the\_garden}\,(\lambda x.\, \mathbf{kiss}\, x\, \mathbf{mary})\,\mathbf{john}$

where $\mathbf{in}$ is of type $\mathbf{e} \to (\mathbf{e} \to \mathbf{t}) \to \mathbf{e} \to \mathbf{t}$.

Using a neo-Davidsonian approach, the semantic interpretation of (4) would be rather different:

(6) $\exists e.\,(\mathbf{kiss}\, e) \wedge (\mathbf{agent}\, e\, \mathbf{john}) \wedge (\mathbf{patient}\, e\, \mathbf{mary}) \wedge (\mathbf{location}\, e\, \mathbf{the\_garden})$

It then appears that the entailment relation existing between (4) and (1) is semantically accounted for by the purely logical entailment of (3) by (6). This is not the case with the traditional Montagovian approach, where the entailment of (2) by (5) would necessitate some meaning postulates.

For the issue at hand in this paper, namely the semantic treatment of *wh*-questions, a neo-Davidsonian approach will allow us to interrogate the different thematic roles using a unique *interrogative quantifier*. Consider the three *wh*-questions one may derive from (4):

(7) a. Who did kiss Mary in the garden?
b. Whom did John kiss in the garden?
c. Where did John kiss Mary?

Using a unique interrogative quantifier, say "WHICH", our semantic account of (7a-c) will amount to a logical translation of the following respective paraphrases:

(8) a. WHICH *is the* agent *of the kissing event whose patient is Mary and whose location is the garden?*
b. WHICH *is the* patient *of the kissing event whose agent is John and whose location is the garden?*
c. WHICH *is the* location *of the kissing event whose agent is John and whose patient is Mary?*

Now, as it will appear in the sequel, this unique interrogative quantifier, WHICH, is in fact the existential quantifier of inquisitive semantics.

### 2.3 Inquisitive semantics

In Montague's intensional logic [20], as in modal logic, a declarative proposition is semantically interpreted as a set of possible worlds. In Hamblin-like logics of questions and answers, a question is identified with its set of possible answers. Therefore, since an answer is itself modeled by a declarative proposition, a question must be modeled by a set of declarative propositions. At the semantic levels, it means that a question must be interpreted as a set of sets of possible worlds.

Inquisitive semantics elaborates on this idea and stipulates, in addition, that both declarative and interrogative propositions must be interpreted as non-emtpty sets of sets of possible worlds that are downward closed by set inclusion. The consequences of this principle are twofold. On the one hand, inquisitive semantics provides a framework in which both declarative and interrogative expressions are treated in a uniform way. It is even the case that there is no neat separation between interrogative and declarative forms. In fact, in inquisitive semantics, every proposition has both an informative and an inquisitive content. On the other hand, interpretating an inquisitive proposition as a set of sets that is downward closed allows conjunction and disjunction to be defined in a standard way, i.e., as intersection and union, respectively. The same is true of quantifiers: universal quantification is interpreted as an arbitrary intersection while existential quantification is interpreted as an arbitrary union. Let us illustrate all this with examples.

Let us posit a domain of interpretation, $D = \{a, b\}$, with two individuals (which we will call *Alice* and *Bob*), and a set of four possible worlds $W = \{\textsc{aa}, \textsc{as}, \textsc{sa}, \textsc{ss}\}$. The intended meaning of these four possible worlds is as follows: $\textsc{aa}$ is the world where both Alice and Bob are awake; $\textsc{as}$ is the world where Alice is awake and Bob is sleeping; $\textsc{sa}$ is the world where Alice is sleeping and Bob is awake; $\textsc{ss}$ is the world where they are both sleeping. Then, the proposition $\varphi_1$ that *Alice sleeps* and the proposition $\varphi_2$ that *Bob sleeps* are interpreted as follows:

$$\begin{aligned}
[\![\varphi_1]\!] &= \{\{\textsc{sa}, \textsc{ss}\}, \{\textsc{sa}\}, \{\textsc{ss}\}, \varnothing\} \\
[\![\varphi_2]\!] &= \{\{\textsc{as}, \textsc{ss}\}, \{\textsc{as}\}, \{\textsc{ss}\}, \varnothing\}
\end{aligned}$$

The inquisitive conjunction of $\varphi_1$ and $\varphi_2$ is interpreted as the intersection of their interpretations:

$$\begin{aligned}
[\![\varphi_1 \wedge \varphi_2]\!] &= [\![\varphi_1]\!] \cap [\![\varphi_2]\!] \\
&= \{\{\textsc{sa}, \textsc{ss}\}, \{\textsc{sa}\}, \{\textsc{ss}\}, \varnothing\} \cap \{\{\textsc{as}, \textsc{ss}\}, \{\textsc{as}\}, \{\textsc{ss}\}, \varnothing\} \\
&= \{\{\textsc{ss}\}, \varnothing\}
\end{aligned}$$

It corresponds to the assertion that both Alice and Bob are sleeping.

The inquisitive disjunction of $\varphi_1$ and $\varphi_2$ is interpreted as the union of their interpretations:

$$\begin{aligned}
[\![\varphi_1 \vee \varphi_2]\!] &= [\![\varphi_1]\!] \cup [\![\varphi_2]\!] \\
&= \{\{\textsc{sa}, \textsc{ss}\}, \{\textsc{sa}\}, \{\textsc{ss}\}, \varnothing\} \cup \{\{\textsc{as}, \textsc{ss}\}, \{\textsc{as}\}, \{\textsc{ss}\}, \varnothing\} \\
&= \{\{\textsc{as}, \textsc{ss}\}, \{\textsc{sa}, \textsc{ss}\}, \{\textsc{as}\}, \{\textsc{sa}\}, \{\textsc{ss}\}, \varnothing\}
\end{aligned}$$

This disjunction does not correspond to a proposition asserting that Alice or Bob is sleeping, but rather to the question *whether Alice or Bob is sleeping*. The mere assertion, $\varphi_3$, that *Alice or Bob is sleeping* is interpreted in a different way:

$$[\![\varphi_3]\!] = \{\{\textsc{as}, \textsc{sa}, \textsc{ss}\}, \{\textsc{as}, \textsc{sa}\}, \{\textsc{as}, \textsc{ss}\}, \{\textsc{sa}, \textsc{ss}\}, \{\textsc{as}\}, \{\textsc{sa}\}, \{\textsc{ss}\}, \varnothing\}$$

The proposition, $\varphi_4$ asserting that *Alice does not sleep* is interpreted as follows:

$$[\![\varphi_4]\!] = \{\{\textsc{aa}, \textsc{as}\}, \{\textsc{aa}\}, \{\textsc{as}\}, \varnothing\}$$

Then, the inquisitive disjunction of $\varphi_1$ and $\varphi_4$ corresponds to the polar question *whether Alice is sleeping*:

$$\begin{aligned}
[\![\varphi_1 \vee \varphi_4]\!] &= [\![\varphi_1]\!] \cup [\![\varphi_4]\!] \\
&= \{\{\textsc{sa}, \textsc{ss}\}, \{\textsc{sa}\}, \{\textsc{ss}\}, \varnothing\} \cup \{\{\textsc{aa}, \textsc{as}\}, \{\textsc{aa}\}, \{\textsc{as}\}, \varnothing\} \\
&= \{\{\textsc{aa}, \textsc{as}\}, \{\textsc{sa}, \textsc{ss}\}, \{\textsc{aa}\}, \{\textsc{as}\}, \{\textsc{sa}\}, \{\textsc{ss}\}, \varnothing\}
\end{aligned}$$

The interpretation of an inquisitive proposition being downward closed, it is completely characterized by its maximal elements. In our last example, these maximal elements are $\{\text{AA}, \text{AS}\}$ and $\{\text{SA}, \text{SS}\}$. They respectively corresponds to the set of worlds where Alice does not sleep, and to the one where she sleeps. In other words, these two maximal elements correspond to two modal propositions: one asserting that *Alice does not sleep*, and the other one, that *Alice sleeps.* These two propositions are precisely the two possible answers to the polar question *whether Alice is sleeping or not.* This illustrates that these maximal elements are, in fact, the counterpart of Hamblin's alternatives.

In inquisitive semantics, a proposition has both an informative and an inquisitive content. Its inquisitive content, which corresponds to Hamblin's alternatives, is given by the maximal elements, while its informative content is given by the merging of these maximal elements. For instance, the informative content of proposition $\varphi_1 \vee \varphi_2$ is that somebody (Alice or Bob) is sleeping, and its inquisitive content is the issue whether Alice or Bob is sleeping. The proposition may then be paraphrased as follows: *knowing that somebody is sleeping, one wonders whether Alice or Bob is sleeping.* The interpretation of a mere assertion such as $\varphi_1$ has only one maximal element. Accordingly, its inquisitive content is trivial, and its paraphrase would be: *knowing that Alice is sleeping, one wonders whether she is sleeping.* Similarly, a mere question such as $\varphi_1 \vee \varphi_4$ has a trivial informative content: *knowing that Alice sleeps or does not sleep, one wonders whether she is sleeping.* This absence of an actual informative content corresponds to the fact that the set of maximal elements of the proposition covers the set of possible worlds.

Inquisitive semantics features two projection operators, ! and ?, that respectively trivialize the inquisitive content and the informative content of a proposition. These operators may be defined as follows:

$$[\![!\varphi]\!] = \mathscr{P}(\bigcup [\![\varphi]\!])$$
$$[\![?\varphi]\!] = [\![\varphi]\!] \cup \mathscr{P}(W \setminus \bigcup [\![\varphi]\!])$$

where $W$ is the set of possible worlds. Then, for any proposition $\varphi$, one has:

$$\varphi = !\varphi \wedge ?\varphi$$

Interestingly enough, these projection operators allow the interpretation of the logical connectives to be refined by providing them with different possible meanings. For instance, proposition $\varphi_3$, which corresponds to a non-inquisitive disjunction of $\varphi_1$ and $\varphi_2$, may be expressed as $!(\varphi_1 \vee \varphi_2)$.

We complete this brief introduction to inquisitive semantics by defining first-order inquisitive logic.

Let $\langle \mathcal{F}, \mathcal{R} \rangle$ be the signature of a first-order language, where $\mathcal{F}$ is the set of function symbols, and $\mathcal{R}$ is the set of relation symbols. From this signature together with a set $\mathcal{X}$ of first-order variables, the notions of terms and of first-order formulas are defined in the standard way.

The notion of a model is as usual in modal logic, i.e., a triple $\langle D, W, \mathcal{I} \rangle$, where $D$ is the domain of interpretation, $W$ is the set of possible worlds, an $\mathcal{I}$ is an interpretation function ranging over $\mathcal{F} \cup \mathcal{R}$, such that:

$$\begin{aligned} &\mathcal{I}(F) \in D^{D^n} && \text{for } F \in \mathcal{F} \text{ of arity } n \\ &\mathcal{I}(R) \in \mathscr{P}(W)^{D^n} && \text{for } R \in \mathcal{R} \text{ of arity } n \end{aligned}$$

Given a valuation $\xi$ from $\mathcal{X}$ into $D$, the interpretation $[\![t]\!]_\xi$ of a term $t$ is defined as usual, and the interpretation of a first-order formula is then given by the following equations:

$$\begin{aligned} [\![R(t_1, \ldots, t_n)]\!]_\xi &= \mathscr{P}(\mathcal{I}(R)([\![t_1]\!]_\xi, \ldots, [\![t_n]\!]_\xi)) \\ [\![\neg\varphi]\!]_\xi &= \{s \mid \forall t \in [\![\varphi]\!]_\xi .\, s \cap t = \varnothing\} \\ [\![\varphi \wedge \psi]\!]_\xi &= [\![\varphi]\!]_\xi \cap [\![\psi]\!]_\xi \\ [\![\varphi \vee \psi]\!]_\xi &= [\![\varphi]\!]_\xi \cup [\![\psi]\!]_\xi \\ [\![\varphi \to \psi]\!]_\xi &= \{s \mid \forall t \subseteq s .\, t \in [\![\varphi]\!]_\xi \to t \in [\![\psi]\!]_\xi\} \\ [\![\forall x .\, \varphi]\!]_\xi &= \textstyle\bigcap_{d \in D} [\![\varphi]\!]_{\xi[x:=d]} \\ [\![\exists x .\, \varphi]\!]_\xi &= \textstyle\bigcup_{d \in D} [\![\varphi]\!]_{\xi[x:=d]} \end{aligned}$$

As for the projection operators ! and ?, they may be added as defined connectives:

$$\begin{aligned} !\varphi &= \neg\neg\varphi \\ ?\varphi &= \varphi \vee \neg\varphi \end{aligned}$$

In order to illustrate the difference between inquisitive first-order logic and usual classical first-order logic let us work out the inquisitive interpretation of an existential formula such as $\exists x .\, \mathbf{sleep}\, x$.

For the purpose of our example, we consider a model with the same domain of interpretation and the same set of possible worlds as previously. Then, we let the interpretation function $\mathcal{I}$ be such that:

$$\begin{aligned} \mathcal{I}(\mathbf{sleep})(a) &= \{\textsc{sa}, \textsc{ss}\} \\ \mathcal{I}(\mathbf{sleep})(b) &= \{\textsc{as}, \textsc{ss}\} \end{aligned}$$

In this setting, we have:

$$\begin{aligned} [\![\exists x .\, \mathbf{sleep}\, x]\!]_\xi &= \textstyle\bigcup_{d \in D} [\![\mathbf{sleep}\, x]\!]_{\xi[x:=d]} \\ &= ([\![\mathbf{sleep}\, x]\!]_{\xi[x:=a]}) \cup ([\![\mathbf{sleep}\, x]\!]_{\xi[x:=b]}) \\ &= (\mathscr{P}(\mathcal{I}(\mathbf{sleep})(a))) \cup (\mathscr{P}(\mathcal{I}(\mathbf{sleep})(b))) \\ &= \{\{\textsc{sa}, \textsc{ss}\}, \{\textsc{sa}\}, \{\textsc{ss}\}, \varnothing\} \cup \{\{\textsc{as}, \textsc{ss}\}, \{\textsc{as}\}, \{\textsc{ss}\}, \varnothing\} \\ &= \{\{\textsc{as}, \textsc{ss}\}, \{\textsc{sa}, \textsc{ss}\}, \{\textsc{as}\}, \{\textsc{sa}\}, \{\textsc{ss}\}, \varnothing\} \end{aligned}$$

This example shows that an inquisitive existential quantification, in general, has both an actual informative and an actual inquisitive content. This double content, which in the present case is the fact that somebody is sleeping and the issue whether it is Alice or Bob, may be adjusted using the projection operators. This gives rise to three kinds of existential quantifications, which in our example correspond to the following formulas:

(9) a. $!\,\exists x.\, \mathbf{sleep}\, x$

b. $?\,\exists x.\, \mathbf{sleep}\, x$

c. $\exists x.\, \mathbf{sleep}\, x$

Intuitively, these three formulas are the logical counterpart the three following utterances, respectively:

(10) a. Somebody is sleeping.

b. Who is sleeping? (*nobody* being a possible answer)

c. Somebody is sleeping. Who is it?

### 2.4 Abstract Categorial Grammars

The account of *wh*-question semantics that we give in the next sections is formalized using the framework of abstract categorial grammars [13]. For the sake of self-containedness, we give here the definitions necessary to understand this formalism.

We assume from the reader some acquaintance with the simply typed $\lambda$-calculus. Nevertheless, in order to fix the terminology, we briefly reminds the main definitions that we will be need in the sequel. In particular, we review the notions of *simple types*, *higher-order signature*, and *linear $\lambda$-terms* built upon a higher-order linear signature.

Let $A$ be a set of atomic types. The set $\mathscr{T}(A)$ of *simple types* built upon $A$ is inductively defined as follows:

1. if $a \in A$, then $a \in \mathscr{T}(A)$;
2. if $\alpha, \beta \in \mathscr{T}(A)$, then $(\alpha \to \beta) \in \mathscr{T}(A)$.

Given two sets of atomic types, $A$ and $B$, a mapping $h : \mathscr{T}(A) \to \mathscr{T}(B)$ is called a *type homomorphism* (or a *type substitution*) if it satisfies the following condition:

$$h(\alpha \to \beta) = h(\alpha) \to h(\beta)$$

A *higher-order signature* consists of a triple $\Sigma = \langle A, C, \tau \rangle$, where:

1. $A$ is a finite set of atomic types;
2. $C$ is a finite set of constants;
3. $\tau : C \to \mathscr{T}(A)$ is a function that assigns to each constant in $C$ a linear implicative type in $\mathscr{T}(A)$.

Let $X$ be a infinite countable set of $\lambda$-variables. The set $\Lambda(\Sigma)$ of *linear $\lambda$-terms* built upon a higher-order linear signature $\Sigma = \langle A, C, \tau \rangle$ is inductively defined as follows:

1. if $c \in C$, then $c \in \Lambda(\Sigma)$;
2. if $x \in X$, then $x \in \Lambda(\Sigma)$;

3. if $x \in X$, $t \in \Lambda(\Sigma)$, and $x$ occurs free in $t$ exactly once, then $(\lambda x.\, t) \in \Lambda(\Sigma)$;
4. if $t, u \in \Lambda(\Sigma)$, and the sets of free variables of $t$ and $u$ are disjoint, then $(t\, u) \in \Lambda(\Sigma)$.

Let $\Sigma_1$ and $\Sigma_2$ be two signatures. We say that a mapping $h : \Lambda(\Sigma_1) \to \Lambda(\Sigma_2)$ is a $\lambda$-term homomorphism if it satisfies the following conditions:

$$h(x) = x$$
$$h(\lambda x.\, t) = \lambda x.\, h(t)$$
$$h(t\, u) = h(t)\, (h(u))$$

Given a higher-order linear signature $\Sigma = \langle A, C, \tau \rangle$, each linear $\lambda$-term in $\Lambda(\Sigma)$ may possibly be assigned a linear implicative type in $\mathscr{T}(A)$. This type assignment obeys the following typing rules:

$$\vdash_\Sigma\ c : \tau_\Sigma(c) \quad \text{(CONS)}$$

$$x : \alpha \vdash_\Sigma\ x : \alpha \quad \text{(VAR)}$$

$$\frac{\Gamma, x : \alpha \vdash_\Sigma\ t : \beta}{\Gamma \vdash_\Sigma\ (\lambda x.\, t) : (\alpha \to \beta)} \ \text{(ABS)}$$

$$\frac{\Gamma \vdash_\Sigma\ t : (\alpha \to \beta) \qquad \Delta \vdash_\Sigma\ u : \alpha}{\Gamma, \Delta \vdash_\Sigma\ (t\, u) : \beta} \ \text{(APP)}$$

where $\mathrm{dom}(\Gamma) \cap \mathrm{dom}(\Delta) = \varnothing$.

Let $\Sigma_1 = \langle A_1, C_1, \tau_1 \rangle$ and $\Sigma_2 = \langle A_2, C_2, \tau_2 \rangle$ be two higher-order signatures. A *lexicon*, $\mathscr{L} : \Sigma_1 \to \Sigma_2$, is defined to be a realization of $\Sigma_1$ into $\Sigma_2$, i.e., an interpretation of the atomic types of $\Sigma_1$ as types built upon $A_2$, together with an interpretation of the constants of $\Sigma_1$ as linear $\lambda$-terms built upon $\Sigma_2$. These two interpretations must be such that their homomorphic extensions commute with the typing relations. More formally, a *lexicon* $\mathscr{L}$ from $\Sigma_1$ to $\Sigma_2$ is defined to be a pair $\mathscr{L} = \langle F, G \rangle$ such that:

1. $F : A_1 \to \mathscr{T}(A_2)$ is a function that interprets the atomic types of $\Sigma_1$ as linear implicative types built upon $A_2$;
2. $G : C_1 \to \Lambda(\Sigma_2)$ is a function that interprets the constants of $\Sigma_1$ as linear $\lambda$-terms built upon $\Sigma_2$;
3. the interpretation functions are compatible with the typing relation, i.e., for any $c \in C_1$, the following typing judgement is derivable:

$$\vdash_{\Sigma_2}\ G(c) : \hat{F}(\tau_1(c)) \qquad (\dagger)$$

where $\hat{F}$ is the unique homomorphic extension of $F$.

Remark that Condition (†) compels $G(c)$ to be typable with respect to the empty typing environment. This means that $G$ interprets each constant $c$ as a closed linear $\lambda$-term. Now, writing $\mathscr{L}$ for both the homomorphic extensions $F$ and $G$, Condition (†) ensures that the following commutation property holds for every $t \in \Lambda(\Sigma_1)$:

$$\text{if} \quad \vdash_{\Sigma_1} t : \alpha \quad \text{then} \quad \vdash_{\Sigma_2} \mathscr{L}(t) : \mathscr{L}(\alpha)$$

We now define an *abstract categorial grammar* (ACG, for short) as a quadruple, $\mathscr{G} = \langle \Sigma_1, \Sigma_2, \mathscr{L}, S \rangle$, where:

1. $\Sigma_1$ and $\Sigma_2$ are two higher-order linear signatures; they are called the *abstract vocabulary* and the *object vocabulary*, respectively;
2. $\mathscr{L} : \Sigma_1 \to \Sigma_2$ is a lexicon from the abstract vocabulary to the object vocabulary;
3. $S$ is an atomic type of the abstract vocabulary; it is called the *distinguished type* of the grammar.

Every ACG $\mathscr{G}$ generates two languages: an *abstract language*, $\mathcal{A}(\mathscr{G})$, and an object language $\mathcal{O}(\mathscr{G})$.

The abstract language, which may be seen as a set of abstract parse structures, is the set of closed linear $\lambda$-terms built upon the abstract vocabulary and whose type is the distinguished type of the grammar. It is formally defined as follows:

$$\mathcal{A}(\mathscr{G}) = \{t \in \Lambda(\Sigma_1) \, : \, \vdash_{\Sigma_1} t \colon S \text{ is derivable}\}$$

The *object language*, which may be seen as the set of surface forms generated by the grammar, is defined to be the image of the abstract language by the term homomorphism induced by the lexicon.

$$\mathcal{O}(\mathscr{G}) = \{t \in \Lambda(\Sigma_2) \, : \, \exists u \in \mathcal{A}(\mathscr{G}). \ t =_{\beta\eta} \mathscr{L}(u)\}$$

Both the abstract language and the object language generated by an ACG are sets of linear $\lambda$-terms. This allows more specific data structures such as strings, trees, or first-order terms to be represented. A string of symbols, for instance, can be encoded as a composition of functions. Consider an arbitrary atomic type $s$, and define $\sigma \triangleq s \to s$ to be the type of strings. Then, a string such as '*abbac*' may be represented by the linear $\lambda$-term:

$$\lambda x.\, a\,(b\,(b\,(a\,(c\,x)))),$$

where the atomic strings '$a$', '$b$', and '$c$' are declared to be constants of type $\sigma$. In this setting, the empty word is represented by the identity function:

$$\epsilon \triangleq \lambda x.\, x$$

and concatenation is defined to be functional composition:

$$\_ + \_ \triangleq \lambda\alpha.\, \lambda\beta.\, \lambda x.\, \alpha\,(\beta\, x),$$

which is indeed an associative operator that admits the identity function as a unit.

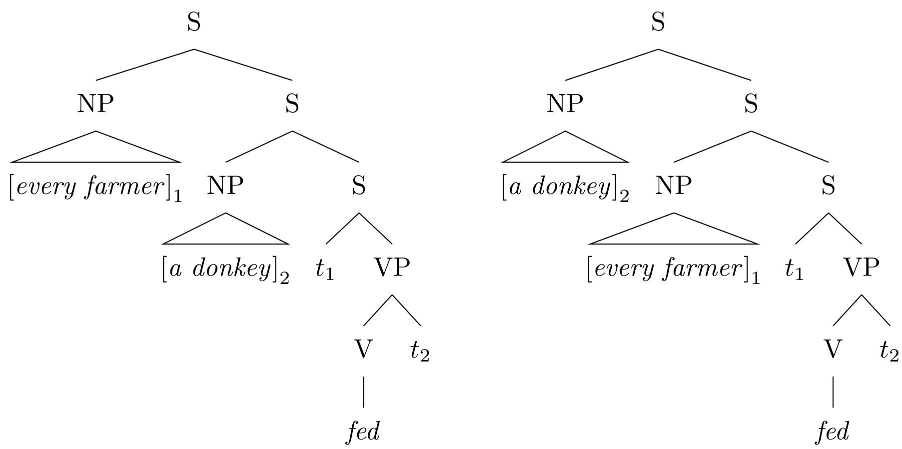
## 3 A categorial formalisation of the syntax and semantics of *wh*-interrogatives

It is usual in the categorial grammar tradition to distinguish between the quantified noun phrases and the mere noun phrases, at the syntactic level. While the latter are assigned a simple atomic type $np$, the former are assigned the functional type $(np \to s) \to s$, which reflects the fact that a quantified expression takes a scope. This allows for a smooth treatment of scope ambiguities.

Let us illustrate this approach by considering the following sentence:

(11) Every farmer fed a donkey.

the two possible readings of which are captured by the following two syntactic structures:




In order to give an abstract categorial account of sentence (11), one may declare abstract constants of the following types:

$$\begin{aligned} \text{FARMER, DONKEY} &: n \\ \text{A, EVERY} &: n \to (np \to s) \to s \\ \text{FED} &: np \to np \to s \end{aligned}$$

Then, the above syntactic structures are encoded in an almost straightforward way by the $\lambda$-terms given in Figure 1.

This categorial treatment of scope ambiguities, which directly derives from Montague [20], might be problematic when the targeted semantic formalism is Davidson's event semantics. It has indeed been argued that Montague's treatment of quantification does not combine smoothly with event semantics. The problem is that, in event semantics, a declarative sentence that is ultimately interpreted as a truth value ($\mathbf{t}$) is first interpreted as a set of events ($\mathbf{v} \to \mathbf{t}$). Then, switching from the latter interpretation of a sentence to the former necessitates an existential-closure operator, which may badly interact with the quantifiers

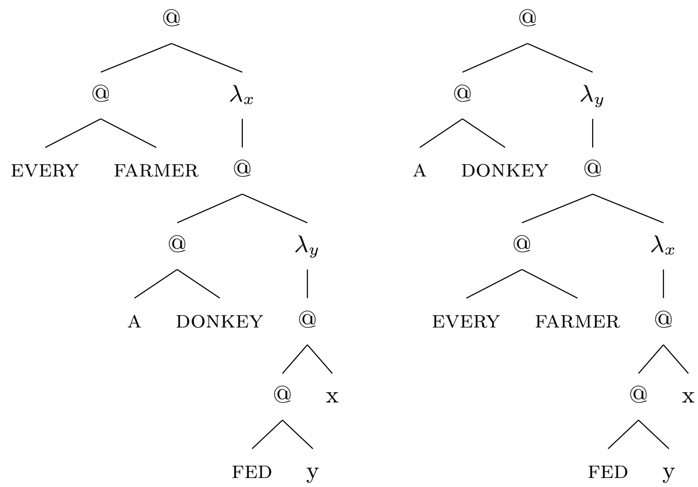



**Fig. 1.** Syntactic structures as linear $\lambda$-terms

that occur in the interpretation of the sentence. Fortunately, the literature provides at least two solutions to this problem. A first one is due to Champollion [5], and a second one to Winter and Zwarts [23]. We follow this second solution since it is in line with the categorial tradition that we are advocating.

Winter and Zwarts' solution consists in assigning two different syntactic types to the sentences. On the one hand, a first type ($s_0$) is used for the "open" sentences, i.e., the sentences that are semantically interpreted as sets of events, and, on the other hand, a second syntactic type ($s$) is used for the sentences that are interpreted as truth values. Then, the existential closure operator allows values of type $s_0$ to be coerced into values of type $s$. Accordingly, the abstract signature we have sketched above is transformed as follows:

$$\begin{aligned} \text{FARMER, DONKEY} &: n \\ \text{A, EVERY} &: n \to (np \to s) \to s \\ \text{FED} &: np \to np \to s_0 \\ \text{E-CLOS} &: s_0 \to s \end{aligned}$$

This ensures that the existential closure operator will always take a narrower scope with respect to the other quantifiers.

Now, it is well known that there is a strong analogy between quantifier raising and *wh*-extraction. Following this analogy suggests that we should assign to a *wh*-noun phrase the type that we assign to a quantified noun phrase. Typically:

$$\text{WHO} : (np \to s) \to s$$

Similarly, a *wh*-determiner must be assigned the same type as a quantificational determiner:

$$\text{WHICH} : n \to (np \to s) \to s$$

Finally, *wh*-adverbs must be assigned the same type as the quantified adverbial modifiers, which are assigned $((s_0 \to s_0) \to s) \to s$ [14]. Accordingly, we have:

$$\text{WHERE} : ((s_0 \to s_0) \to s) \to s$$

Putting everything together, we end up with an abstract syntax specified by the signature given in Table 2, where E-CLOS and Q are syntactically silent operators. The first one allows an open sentence to be turned into a closed one. The second allows a declarative proposition to be turned into a polar question.

**Table 2.** Abstract syntax signature

| *Abstract Syntax* | |
|---:|:---|
| FARMER, DONKEY, MEADOW : | $n$ |
| THE : | $n \to np$ |
| A, SOME, EVERY, WHICH : | $n \to (np \to s) \to s$ |
| IN : | $np \to s_0 \to s_0$ |
| FED, DID-FEED : | $np \to np \to s_0$ |
| WHO : | $(np \to s) \to s$ |
| WHERE : | $((s_0 \to s_0) \to s) \to s$ |
| E-CLOS : | $s_0 \to s$ |
| Q : | $s \to s$ |

In abstract categorial grammars, the language generated by an abstract signature (such as the one given in Table 2) acts as a pivot language between surface forms and semantic interpretations. This is typically the way an abstract categorial grammar models the syntax-semantics interface:

$$\begin{tikzcd}
 & \textit{Abstract Syntax} \arrow[dl, "\text{Surface Realization}"'] \arrow[dr, "\text{Semantics Interpretation}"] & \\
\textit{Surface Forms} & & \textit{Semantics}
\end{tikzcd}$$

Consequently, in order to complete the picture, it remains to give the syntactic and the semantic translations of the abstract syntax specified by the signature of Table 2. These are given in Table 3 and Table 4, respectively.

We may now illustrate the overall approach by treating the following example:

(12) Where did every farmer feed a donkey?

The above sentence contains three binding expression: a *wh*-adverb (*where*), and two quantified noun phrases (*every farmer* and *a donkey*). The relative scope of

**Table 3.** Surface realisation lexicon

| *Surface Realization* | |
|---|---|
| FARMER | $:= \boldsymbol{farmer}$ |
| DONKEY | $:= \boldsymbol{donkey}$ |
| MEADOW | $:= \boldsymbol{meadow}$ |
| THE | $:= \lambda x.\, \boldsymbol{the} + x$ |
| A | $:= \lambda x p.\, p\,(\boldsymbol{a} + x)$ |
| SOME | $:= \lambda x p.\, p\,(\boldsymbol{some} + x)$ |
| EVERY | $:= \lambda x p.\, p\,(\boldsymbol{every} + x)$ |
| WHICH | $:= \lambda x y.\, \boldsymbol{which} + x + (y\, \epsilon)$ |
| IN | $:= \lambda x y.\, y + \boldsymbol{in} + x$ |
| FED | $:= \lambda x y.\, y + \boldsymbol{fed} + x$ |
| DID-FEED | $:= \lambda x y.\, \boldsymbol{did} + y + \boldsymbol{feed} + x$ |
| WHO | $:= \lambda x.\, \boldsymbol{who} + (x\, \epsilon)$ |
| WHERE | $:= \lambda q.\, \boldsymbol{where} + (q\,(\lambda x.\, x))$ |
| E-CLOS | $:= \lambda x.\, x$ |
| Q | $:= \lambda x.\, x$ |

**Table 4.** Semantic interpretation lexicon

| *Semantic Interpretation* | |
|---|---|
| FARMER | $:= \lambda x.\, \mathbf{farmer}\, x$ |
| DONKEY | $:= \lambda x.\, \mathbf{donkey}\, x$ |
| MEADOW | $:= \lambda x.\, \mathbf{meadow}\, x$ |
| THE | $:= \lambda p.\, \mathbf{the}\,(\lambda x.\, p\, x)$ |
| A, SOME | $:= \lambda p q.\, !(\exists x.\, (p\, x) \wedge (q\, x))$ |
| EVERY | $:= \lambda p q.\, \forall x.\, (p\, x) \rightarrow (q\, x)$ |
| WHICH | $:= \lambda p q.\, \exists x.\, (p\, x) \wedge (q\, x)$ |
| IN | $:= \lambda x p.\, \lambda e.\, (p\, e) \wedge (\mathbf{location}\, e\, x)$ |
| FED, DID-FEED | $:= \lambda x y.\, \lambda e.\, (\mathbf{fed}\, e) \wedge (\mathbf{patient}\, e\, x) \wedge (\mathbf{agent}\, e\, y)$ |
| WHO | $:= \lambda p.\, \exists x.\, p\, x$ |
| WHERE | $:= \lambda p.\, \exists x.\, p\,(\lambda q.\, \lambda e.\, (q\, e) \wedge (\mathbf{location}\, e\, x))$ |
| E-CLOS | $:= \lambda p.\, !(\exists e.\, p\, e)$ |
| Q | $:= \lambda x.\, ?x$ |

these binding expressions must obey the constraint that the *wh*-expression takes the widest scope. Consequently, we are only left with two possible readings: one where the relative scope of the quantified noun phrases follows the surface order; another one where *a donkey* takes scope over *every farmer*.

These two readings correspond to two different syntactic structures, which are modelled as $\lambda$-terms built upon the signature given in Table 2. These $\lambda$-terms are the following ones:

(13) $\textsc{q}\,(\textsc{where}$
$(\lambda f.\, \textsc{every farmer}$
$(\lambda x.\, \textsc{a donkey}\,(\lambda y.\, \textsc{e-clos}\,(f\,(\textsc{did-feed}\, y\, x))))))$

(14) $\textsc{q}\,(\textsc{where}$
$(\lambda f.\, \textsc{a donkey}$
$(\lambda x.\, \textsc{every farmer}\,(\lambda y.\, \textsc{e-clos}\,(f\,(\textsc{did-feed}\, x\, y))))))$

Both (13) and (14) yield the same result when the surface realization lexicon given in Table 3 is applied to them:

(15) $\boldsymbol{where} + \boldsymbol{did} + \boldsymbol{every} + \boldsymbol{farmer} + \boldsymbol{feed} + \boldsymbol{a} + \boldsymbol{donkey}$

By contrast, when the semantic interpretation given in Table 4 is applied to them, (13) and (14) yield the two expected different readings:

(16) $?\,\exists x.\, \forall y.\, (\mathbf{farmer}\, y) \rightarrow$
$!\,(\exists z.\, (\mathbf{donkey}\, z) \wedge$
$!\,(\exists e.\, (\mathbf{fed}\, e) \wedge (\mathbf{patient}\, e\, z) \wedge (\mathbf{agent}\, e\, y) \wedge (\mathbf{location}\, e\, x)))$

(17) $?\,\exists x.\, !\,(\exists y.\, (\mathbf{donkey}\, y) \wedge$
$(\forall z.\, (\mathbf{farmer}\, z) \rightarrow$
$!\,(\exists e.\, (\mathbf{fed}\, e) \wedge (\mathbf{patient}\, e\, y) \wedge (\mathbf{agent}\, e\, z) \wedge (\mathbf{location}\, e\, x))))$

## 4 Controlling *wh*-extraction and quantifier raising

The grammar we have sketched in the previous section is quite simple, and has the advantage of highlighting the parallel that exists between declarative and interrogative sentences. In particular, it is based on a uniform treatment of quantification raising and *wh*-extraction. This simplicity, however, is not without its drawbacks. These are threefold. Firstly, our grammar assigns the same syntactic categories to both the declarative and the interrogative forms (for instance, EVERY and WHICH are both assigned $n \rightarrow (np \rightarrow s) \rightarrow s$). This gives rise to a grammar that generates ungrammatical surface forms such as:

(18) *Every farmer fed which donkey in which meadow.

Secondly, allowing the quantifiers to take any possible scope results in spurious ambiguities. For instance, a sentence such as:

(19) A farmer fed a donkey in a meadow.

will give rise to six different abstract syntactic structures the semantic interpretations of which are all logically equivalent. Finally, the interactions between *wh*-extraction and quantifier raising must obey some constraints. For instance, in a *wh*-question, the *wh*-quantifier must always take the wider scope. Consider again example (12). The signature given in Table 2 allows one to built syntactic structures, such as the following one, that do not respect the *wh*-quantifier wider scope constraint:

(20) $\textsc{every farmer}$
$$(\lambda x.\, \textsc{where}\, (\lambda f.\, \textsc{a donkey}\, (\lambda y.\, \textsc{e-clos}\, (f\, (\textsc{did-feed}\, y\, x)))))$$

Consequently, our grammar might allow nonsensical semantic interpretations to be derived.

The three kinds of defects that our grammar presents are all the consequence of a same fact: the *abstract syntax* signature of Table 2 allows too many abstract syntactic structures to be derived. In order to overcome this difficulty, we should be able to select among the λ-terms that can be built upon the signature of Table 2 the ones that correspond to legitimate abstract syntactic structures. A modular and efficient solution to this problem consists in using an additional abstract categorial grammar (which we will call the *control grammar*) in order to rule out the illegitimate abstract syntactic structures. This idea results in the following grammatical architecture:

$$\begin{tikzcd}
 & \textit{Control Signature} \arrow[d, "\text{Control Lexicon}"] & \\
 & \textit{Abstract Syntax} \arrow[dl, "\text{Surface Realization}"'] \arrow[dr, "\text{Semantics Interpretation}"] & \\
\textit{Surface Forms} & & \textit{Semantics}
\end{tikzcd}$$

In this architecture, the control signature may be seen as a type refinement of the abstract syntax. Typically, it distinguishes between different types of verb phrases and sentences, e.g., declarative verb phrase (*vp*) and interrogative verb phrase (*vpq*). This is useful, for instance, to prevent the grammar from assigning a declarative meaning to an interrogative sentence. It also distinguishes between different types of noun phrases, e.g, existentially quantified noun phrases (*npe*) and universally quantified noun phrases (*npu*). This is used to prevent a quantified noun phrase to be raised over another quantified noun phrase of the same quantificational force.

We have developed such a control grammar, which is unfortunately too large to be presented here[1]. Just to give a flavor of it, by way of illustration, we give in Tables 5 and 6 the excerpt that allows example (12) to be dealt with.

**Table 5.** Control signature (excerpt)

| *Control Signature* | |
|---:|:---|
| a : | $n \to npe$ |
| every : | $n \to npu$ |
| donkey : | $n$ |
| farmer : | $n$ |
| Inpe : | $npe \to np$ |
| Inpu : | $npu \to np$ |
| did-feed : | $np \to vpq$ |
| did-feed$_2$ : | $vpq_{np}$ |
| where : | $sq_0 \to s$ |
| SQ4 : | $np \to vpq \to sq_0$ |
| SQ14 : | $npu \to vpq_{np} \to sq_{0npe}$ |
| QRq13 : | $npe \to sq_{0npe} \to sq_0$ |

**Table 6.** Control lexicon (excerpt)

| *Control Lexicon* | |
|---:|:---|
| a := | $\textsc{a}$ |
| every := | $\textsc{every}$ |
| donkey := | $\textsc{donkey}$ |
| farmer := | $\textsc{farmer}$ |
| Inpe := | $\lambda x.\, x$ |
| Inpu := | $\lambda x.\, x$ |
| did-feed := | $\lambda pxf.\, p\,(\lambda y.\, f\,(\textsc{did-feed}\, y\, x))$ |
| did-feed$_2$ := | $\lambda xyf.\, f\,(\textsc{did-feed}\, x\, y)$ |
| where := | $\lambda p.\, \textsc{q}\,(\textsc{where}\,(\lambda f.\, p\,(\lambda s.\, \textsc{e-clos}\,(f\, s))))$ |
| SQ4 := | $\lambda pqf.\, p\,(\lambda x.\, q\, x\, f)$ |
| SQ14 := | $\lambda pqxf.\, p\,(\lambda y.\, q\,(x\, y)\, f)$ |
| QRq13 := | $\lambda pqf.\, p\,(\lambda x.\, q\, x\, f)$ |

[1] It is about three times larger than the grammar presented in Section 3.

## 5 Conclusion and future work

The semantic analysis of *wh*-questions that we have presented in this paper is based on a strong parallel between *wh*-words and thematic roles. This parallel, however, is not always as obvious as it might seems. Let us discuss the cases of some *wh*-words that might be problematic: *whose*, *how*, *what*, and *why*.

*whose* raises the problem of modeling the possessive relation. It is well known indeed that the possessive relation is multiple and that it does not correspond to a single thematic relation. In many cases, the type of the relation and the thematic role played by the possessor can be determined from the lexical semantics of the possessed entity. In some other cases, however, the nature of the possessive relation also depends on the nature of the possessor. These difficulties, in fact, are not specific to the use of the *wh*-word *whose*. They are related to the semantics of possessives, which is a subject on its own.

Regarding *how*, the difficulty is also contextual: the meaning of the *wh*-word *how* depends on the expression *how* is paired with. Consider the difference between *how long* and *how far*. In the first case, the interrogated thematic role might be time-related, while in the second case, it is location-related.

In many cases, *what* appears to be close in behavior either to *who* and *whom*, or to *which*. The difference between *what* and *which* seems to come from pragmatic considerations: the interpretation of *what* hugely depends on the context in which this interrogative word is used, while *which* is restrained in its interpretation by the definition of the set from which the choice of the response is made. *what* may also occur in a generic question such as *what did the farmer do*. This question does not interrogate a thematic role but rather the nature of an event. It could be paraphrase as *of which kind of event was the farmer the agent*. In the current state of our model, this cannot be treated because it would require a second-order quantification.

Finally, the difficulty with *why* is that it does not interrogate a thematic role but rather the argument of a discourse relation. Consequently, in order to propose a treatment of *why*, we would need to extend our model with a theory of discourse, including a theory of discursive relations.

All the possible extensions that we have discussed above will be subject to further work.

## References

1. Aarts, B., Chalker, S., Weiner, E.: The Oxford dictionary of English grammar. Oxford University Press (2014)
2. Bender, E.M., Lascarides, A.: Linguistic fundamentals for natural language processing ii: 100 essentials from semantics and pragmatics. Synthesis Lectures on Human Language Technologies **12**(3), 1–268 (2019)
3. Bonial, C., Corvey, W., Palmer, M., Petukhova, V.V., Bunt, H.: A hierarchical unification of lirics and verbnet semantic roles. In: 2011 IEEE Fifth International Conference on Semantic Computing. pp. 483–489. IEEE (2011)

4. Bunt, H.: The dit++ taxonomy for functional dialogue markup. In: AAMAS 2009 Workshop, Towards a Standard Markup Language for Embodied Dialogue Acts. pp. 13–24 (2009)
5. Champollion, L.: The interaction of compositional semantics and event semantics. Linguistics and Philosophy **38**(1), 31–66 (2015)
6. Ciardelli, I., Groenendijk, J., Roelofsen, F.: Inquisitive semantics. Oxford University Press (2018)
7. Ciardelli, I., Roelofsen, F., Theiler, N.: Composing alternatives. Linguistics and Philosophy **40**(1), 1–36 (2017)
8. Dowty, D.R.: Thematic roles and semantics. In: Annual Meeting of the Berkeley Linguistics Society. vol. 12, pp. 340–354 (1986)
9. Fillmore, C.J., Fillmore, S.: Case grammar. Universals in Linguistic Theory. Holt, Rinehart, and Winston: New York pp. 1–88 (1968)
10. Ginzburg, J., Sag, I.: Interrogative investigations. Stanford: CSLI publications (2000)
11. Groenendijk, J., Stokhof, M.: Questions. In: van Benthem, J., ter Meulen, A. (eds.) Handbook of Logic and Language. Elsevier, second edn. (2011)
12. Groenendijk, J., Stokhof, M.: Studies on the Semantics of Questions and the Pragmatics of Answers. Ph.D. thesis, Univ. Amsterdam (1984)
13. de Groote, Ph.: Towards Abstract Categorial Grammars. In: Association for Computational Linguistics, 39th Annual Meeting and 10th Conference of the European Chapter, Proceedings of the Conference. pp. 148–155 (2001)
14. de Groote, Ph., Winter, Y.: A type-logical account of quantification in event semantics. In: Murata, T., Mineshima, K., Bekki, D. (eds.) LENLS, JURISIN, and GABA, Kanagawa, Japan, Revised Selected Papers. Lecture Notes in Computer Science, vol. 9067, pp. 53–65. Springer (2014)
15. Gruber, J.S.: Studies in lexical relations. Ph.D. thesis, Massachusetts Institute of Technology (1965)
16. Hamblin, C.L.: Questions in Montague English. Foundations of Language **10**(1), 41–53 (1973)
17. Kratzer, A., Heim, I.: Semantics in generative grammar, vol. 1185. Blackwell Oxford (1998)
18. Montague, R.: English as a formal language. In: et al., B.V. (ed.) Linguaggi nella Società e nella Tecnica. Edizioni di Communità, Milan (1970), reprinted: [21, pages 188–221]
19. Montague, R.: Universal grammar. Theoria **36**, 373–398 (1970), reprinted: [21, pages 222–246]
20. Montague, R.: The proper treatment of quantification in ordinary english. In: Hintikka, J., Moravcsik, J., Suppes, P. (eds.) Approaches to natural language: proceedings of the 1970 Stanford workshop on Grammar and Semantics. Reidel, Dordrecht (1973), reprinted: [21, pages 247–270]
21. Montague, R.: Formal Philosophy: selected papers of Richard Montague, edited and with an introduction by Richmond Thomason. Yale University Press (1974)
22. Parsons, T.: Events in the Semantics of English, vol. 5. MIT press Cambridge, MA (1990)
23. Winter, Y., Zwarts, J.: Event semantics and Abstract Categorial Grammar. In: Kanazawa, M., Kornai, A., Kracht, M., Seki, H. (eds.) Proceedings of Mathematics of Language, MOL12. Lecture Notes in Artificial Intelligence, LNAI, vol. 6878, pp. 174–191. Springer-Verlag, Berlin (2011)

24. Wiśniewski, A.: Semantics of questions. In: Lappin, S., Fox, C. (eds.) The Handbook of Contemporary Semantic Theory, chap. 9, pp. 271–313. John Wiley & Sons, Ltd (2015)