# Dependencies, semantic constraints and conceptual closeness in a dynamic frame theory

**Abstract.** Results from psycholinguistics and neuroscience, in particular based on the N400, provide ample evidence that semantic relations between lexical items play an important and prominent role during the (semantic) processing of sentences in the brain. For example, although neither *John squeezed an orange* nor *John squeezed an apple* contain a semantic anomaly, they are processed differently in the brain, because *orange* is more expected as the direct object of *squeeze* than *apple*.
In this article we will take a first step in closing this gap between neuroscience and formal semantics. Using frame theory [Löb14] in which lexical items like 'orange' are interpreted as sets of properties, makes it possible to apply strategies both from Dependence Logic [Vää07] and theories of belief revision ([GP92], [Bou98]). In particular, it is possible (i) to define dependency relations between different properties of an object and (ii) to define quantitative plausiblity relations ($\kappa$-rankings) on a frame that determine how this frame is revised or updated with new information.

**Keywords:** formal vs. cognitive semantics, lexical semantic relations, frame theory, dependence logic, belief revision, belief update, $\kappa$-rankings

## 1 Introduction

According to many, if not most, current formal semantic theories, common nouns like 'orange' or 'paper' are basically analyzed as sets of objects. For example, 'orange' is first translated as the lambda-term $\lambda x.orange(x)$, which, in a second step, is interpreted as a subset of the domain, or, more precisely, as a function from this domain to the set of truth values (1-a). Similarly, using an event-based approach, verbs like 'run' are interpreted as sets of events or the corresponding characteristic function (1-b).

(1) a. $[[orange]]^{\mathcal{M}} = \lambda x \in D_{\langle e_{object} \rangle} f_{orange}(x) = 1$
b. $[[run]]^{\mathcal{M}} = \lambda e \in D_{\langle e_{event} \rangle} f_{run}(e) = 1$

In recent years, such approaches to defining the semantics of basic lexical items like common nouns and verbs have been criticized from neuroscience. According to [BH11], those theories are 'by design insensitive to differences between words of the same syntactic category denoting objects of the same type' [BH11, 1343]. As a consequence, they are inappropriate as a theory of semantic processing in the brain. This criticism is based on empirical results from neurophysiological and neuroimaging phenomena like the N400 (For details on this component, see [BH11]), which is a component of event-related potentials (ERP's), whose amplitude is modulated by semantic complexity.

**N400**. Consider the examples in (2) and (3).

(2) a. Jenny put the sweet in her mouth after the lesson.
b. Jenny put the sweet in her pocket after the lesson.

(3) Every morning John makes himself a glass of freshly squeezed juice. He keeps his refrigerator stocked with (oranges/apples/carrots).

A formal semantic analysis of the sentences in (2) differs only in the sort of object assigned to the locative argument of the verb *put*: *mouth* versus *pocket*. Yet, when this sentence is uttered in a context where Jenny leaves the classroom after a lesson, there is a difference in the amplitude of the N400 between *mouth* and *pocket*, showing that there is a difference during processing in the brain that needs to be accounted for by formal semantic theories.

Sentences like (3) were used by [FK99] in an ERP experiment also targeting the N400. The authors found an increasing N400 effect with the ordering 'oranges' $<$ 'apples' $<$ 'carrots'. According to one interpretation of the N400, this effect is closely related to predicting upcoming words in a sentence which is based on semantic relations between words in the memory component of the brain. For example, in (3) both 'apple' and 'carrot' trigger a larger N400 compared to 'orange' because the former are semantically less related to an event of squeezing a fruit than the latter.

**Stimulus subject perception verbs.** Perception-based verbs (henceforth PBVs) refer to sensory properties of objects like 'taste' or 'sound'. Correlated to each sense modality is a set of values that this property can take and which are specific to it. For example, for the property 'sight' appropriate values are 'square', 'oblong' and 'oval'. PBVs admit of a direct-sensory use in which a predicative complement is added.

(4) This melon sounds muffled/tastes sweet/smells fruity.

In addition to the direct perception use, PBVs can be used inferentially. In this case the predicative complement does *not* determine a value of the scale corresponding to the modality expressed by the verb, but a value belonging to another modality.

(5) a. This melon sounds ripe/old/*oval.

This example shows that the inferential use is not always admissible but depends on the types of sense modalities expressed by the verb and the predicative complement. Thus, similar to the examples of simple sentences, semantic processing of this use of PBVs involves semantic relations. In this case, these are relations between different properties of objects that can be changed by actions or events.

**The *the* ... *the* construction.** The third and final construction involving semantic relations discussed in this article is the *the* ... *the*-construction. Similar to the inferential use of PBVs, this construction expresses a dependency relation between the values of two properties over time.

(6) a. The older a stamp, the more expensive it is.
b. The more alcohol you drink, the higher is your blood alcohol concentration.

## 2 Outline of the theory

According to Baggio and Hagoort (2011:1342), formal semantic theories which describe how words belonging to different syntactic categories or denoting different sorts of objects combine to more complex units are 'by design insensitive to differences between words of the same syntactic category denoting objects of the same type'. The authors put the blame for this 'insensitivity' on the fact that such theories focus on truth conditions, i.e. how language relates to the world, and not on considering natural language as a psychological phenomenon.

What is completely missing from the view of information encapsulated both in static and dynamic approaches to meaning in natural language is the aspect that (declarative) sentences describe situations in the world. Such a description can either concern the fact that some property of an object holds (or fails to hold) or that an event (action) occurs which changes some property of an object. One area in which this type of information is dealt with are theories of belief revision and belief update. Belief revision is usually taken as dealing with incorporating new information about a static, unchanging world. By contrast, belief update is about incorporating information about changes in the world that are triggered by actions or events. New information about a static world is incorporated into a ranked belief set (often called an epistemic state). As a consequence, the way such an epistemic state is changed not only depends on the formulas that currently form the belief set (or knowledge base) of an agent but also on the way those formulas (or the possible worlds used to interpreted those formulas) are ranked. Such information cannot be inferred if the meaning is restricted to sortal information, say it is an orange or a running, and if the dynamics only captures discourse information.

The conclusion that we draw from this failure of current formal semantic theories is that semantic processing cannot solely be based on (i) truth-conditional content and (ii) discourse information in form of information about anaphoric relations which leads to the notion of a context change potential in terms of discourse referents or pegs and (iii) (possibly) world knowledge and context information. In addition, there are at least three further types of information: (i) information about the semantic closeness between nominal and verbal concepts, which expresses degrees of expectancy or plausibility between these two types of concepts. This type of information corresponds to ranking functions in theories of belief revision and belief update; (ii) dependency relations between the values of two properties of an object which can be expressed in Dependence Logic and (iii) information about the way such dependencies are related over time if the values of the corresponding properties are changed by events. Such information requires the use of various ranking functions that not only consider static semantic relations but also the way of how such relations can be defined in the context in which not only a static world but a world in which events bring about changes is taken into consideration.

Consider the following example. When processing a common noun like 'orange', a language user only gets sortal information: it is an object of sort 'orange' belonging to a particular subset of the universe (or the domain of the model). This kind of information is exactly what is usually captured in an (extensional, type-theoretic) truth-conditional semantics and which is formalized by the meaning or satisfaction clauses in (1). This aspect of meaning will be called the *proper* or *lexical* meaning of a common noun or an intransitive verb. Thus, as in model-theoretic semantics, the *lexical* meaning of common nouns and verbs is defined in terms of only sortal information and (possibly) its arity. Given *only* this information, no information about non-sortal properties is supplied. In order to get such information, a language user applies both local contextual information and global world knowledge to extend this lexical information, e.g. by information about properties of objects.

| | **sort** | **color** | **form** | **origin** | **ripeness** | **taste** |
|---|---|---|---|---|---|---|
| object | orange | green | oval | spain | ripe | smooth |

**Table 1.** Tabular representation of the lexical meaning of the common noun 'orange' enriched with contextual information and world knowledge

From a linguistic point of view, the representation in Table 1 provides a *decompositional* analysis of a common noun.

(7) $\lambda x(orange(x) \wedge color(x) = green \wedge form(x) = oval \wedge origin(x) = spain \wedge ripeness(x) = ripe \wedge taste(x) = smooth \ldots)$

However, such a decompositional representation of the meaning of a lexical item is still both a flat and completely static structure in the following sense. First, no distinction is made between admissible values for a particular property. Although these values can be ordered (e.g. say in form a scale, i.e. a partially or linearly ordered set), there is no relation that orders them with respect to plausibility or expectancy. Second, no distinction is made between admissible values for objects to which this object can be related. For example, for the denotation of common nouns: what are the most plausible (expected) events that bring about a change w.r.t. one of its properties? Conversely, for events denoted by verbs: with respect to which sorts of objects does the event most likely bring about a change? Third, 'Does the event have more than one outcome, i.e. it is deterministic or non-deterministic?' Fourth, no information about dependencies between (the values of) properties is expressed. Thus, the problem is *not* only related to getting *more* information, but also to the question of *how* this information is ranked and what dependencies exist between different properties. However, in order to impose both expectancy and dependency constraints a decompositional analysis of the denotions of common nouns and verbs is needed because only then is it possible to explicitly refer to the properties with respect to which those constraints are defined.

Another way of looking at this problem is in terms of the information state of a language user. We follow dynamic approaches and define an information state as a set of possibilities consisting of the alternatives that are still open according to the information available to the language user. Consider (7) again. The information state of a language user w.r.t. to this information is given by a set of possible worlds capturing his epistemic uncertainty, which is due to the fact that his knowledge about the values of properties of an orange is only partial and incomplete. As an effect, his knowledge consists of all those possibilities that are compatible with his current knowledge. In the present case the alternatives concern possible expansions of his knowledge about the orange. He then assumes that the actual (correct) description is some subset $U$ of the set $W$ of possible worlds. However, since all possible worlds are assumed to have equal status for the language user, no world is preferred or more expected than any other in the set of all possibilities. As a consequence, updating amounts to intersecting. A further problem concerns the information that a language user can infer from his current information state provided, say, by applying the information provided by the lexical meaning plus context information together with world knowledge. If his information state is a flat structure in the sense that all worlds are taken as equal, no information about the values of properties about which no information is provided can (defeasibly) be drawn. By contrast, if a language user has information both about dependency and expectancy relations, he can use this type of information to (defeasibly or non-defeasibly) infer other pieces of information about the situation described by the sentences he is currently processing. Thus, the cognitive significance of dependency and expectancy relations consists in the fact that given part of a sentence, a language user will defeasibly infer as much additional information about the situation described by the sentence as possible. Consider the following example. Suppose there is an input state representing mostly ducks (say, because the topic of a conversation are ducks). Then an event of swimming is more expected than events of jumping or walking. By contrast, if the topic is about deers, swimming is less expected than jumping.[1]

## 3 Outline of the formalization

### 3.1 Structures for events, objects and their properties

We start by fixing models for objects and events that capture sortal information which is used in defining the lexical meanings of common nouns and verbs.

[1] See [vEvSZB10] for empirical evidence based on an EEG study and references cited therein.

**Definition 1 (Object structure)** *Let $CN$ be a set of object sort symbols like 'orange'. An object structure $\mathcal{O}$ is a quadruple $\langle O, \{P_{cn}\}_{cn \in CN}, \sqsubseteq_o, \sqcup_o \rangle$ s.t. (i) $O$ is a non-empty set of objects; (ii) each $P_{cn}$ is a subset of $O$; (iii) $\sqsubseteq_o$ is the material part-of relation on $O$, which is required to be a partial order and (iv) $\sqcup_o$ is the join operation on $O$, which is required to be a join-semilattice.*

**Definition 2 (Event structure)** *Let $Verb$ be a set of event sort symbols like 'squeeze'. An event structure $\mathcal{E}$ is a quadruple $\langle E, \{P_v\}_{v \in VERB}, \sqsubseteq_e, \sqcup_e \rangle$ s.t. (i) $E$ is a non-empty set of events; (ii) each $P_v$ is a subset of $E$; (iii) $\sqsubseteq_e$ is the material part-of relation on $E$, which is required to be a partial order and (iv) $\sqcup_e$ is the join operation on $E$, which is required to be a join-semilattice.*

Elements of $E$ and $O$ will be called *entities*. At the level of $\mathcal{O}$ and $\mathcal{E}$, entities are taken as elements of the underlying domain of some fixed global model $\mathcal{M}$, which can have parts. This relation is represented by a part-of relations $\sqsubseteq_o$ and $\sqsubseteq_e$, respectively. In addition, they can be 'summed' to form plural entities. This is modeled by the join operations $\sqcup_o$ and $\sqcup_e$, respectively.

What is missing at this level is the view of an entitiy as a 'bundle' of properties, corresponding to a decompositional analysis at the linguistic level. Such a view makes it possible to impose constraints on (the values of) properties of entities denoted by common nouns and events. Properties of objects like 'Ripeness', 'Sound' or 'Age' are represented by partially or linearly ordered sets, called *scale structures.*

**Definition 3 (Scale structure)** *A scale structure $\mathcal{D}$ is a pair $\langle \Delta, \leq \rangle$ s.t. $\Delta$ is a non-empty set of degrees, the set of admissible values for the scale, and $\leq$ is an ordering on $\Delta$, usually either a partial or a linear order. Scales are required to have a least element, which is denoted by $\top$. Intuitively, $\top$ means that no information about the value is known or provided.*

Let $PROP$ be a set of property symbols like 'sort' or 'ripeness' and let $\{\mathcal{D}_p\}_{p \in PROP}$ be a family of scale structures indexed by elements from $PROP$. Elements of $O$ are assigned a subset of $\{\mathcal{D}_p\}_{p \in PROP}$ by a (subset of a) family of partial functions $\{\gamma_p\}_{p \in PROP}$, which assign to an $o \in O$ the scale structure $\mathcal{D}_p$, if defined. The following condition is imposed on this assignment. If $o, o' \in P_{cn}$, then $\gamma_p(o)$ is defined iff $\gamma_p(o')$ is defined and one has $\gamma_p(o) = \gamma_p(o')$, i.e. objects belonging to the same sort are assigned the same scale structures. If $\gamma_p(o)$ is defined for an object of sort $cn$, the property $p$ is *admissible* for objects of sort $cn$.

While processing a common noun, context information and world knowledge provide a language user with the current values of some of the properties assigned to the object denoted by the common noun. This decomposition can be represented as a (finite) conjunction of the form (8).

(8) $\phi_\sigma \wedge \phi_1 \wedge \ldots \wedge \phi_n \; (= \phi)$

In (8), $\phi_\sigma$ expresses sortal information (lexical meaning), i.e. information about the property 'Sort' and the $\phi_i$ non-sortal information (context information and world knowledge), e.g. information about properties like 'Ripeness. Since in general a language user doesn't know the values of all properties of the object, he is epistemically uncertain about the exact 'status' of the object. For example, suppose that w.r.t. a particular melon the values of the properties 'Form' and 'Origin' are known by a language user and that there are exactly two other properties 'Sound' and 'Ripeness', whose possible values are 'dull' or 'muffled' and 'not ripe' or 'ripe', respectively. The set of possibilities can be represented by the following set of assignments. The 'real' melon could be any of the four melons, each corresponding to a variable assignment.

| **object** | **sort** | **form** | **origin** | **sound** | **ripeness** |
|---|---|---|---|---|---|
| $m_1$ | melon | oblong | spain | dull | ripe |
| $m_2$ | melon | oblong | spain | dull | not ripe |
| $m_3$ | melon | oblong | spain | muffled | ripe |
| $m_4$ | melon | oblong | spain | muffled | not ripe |

**Table 2.** A set of possibilities for an object denoted by the common noun 'melon'

### 3.2 Dependence logic

One way of looking at Table 2 is as a table in a database. In Dependence Logic [Vää07], such tables are an instance of a *team*. A team is a set of agents, with an agent being defined as a function from finite sets (or tuples) of variables, called the domain of the agent, into an arbitrary set, called the codomain of the agent. In the present context, agents are objects, i.e. elements of the domain $O$, viewed as bundles of properties.

**Definition 4 (Team Dependence Logic)** *Let $\langle x_1, \ldots, x_n \rangle$ be a finite tuple of property variables such that no two variables are of the same property sort (i.e. each variable has associated with it a sort $p \in PROP$). Let $M$ be the union of the domains $\Delta$ from $\{D_p\}_{p \in PROP}$. An agent is any function from $\langle x_1, \ldots, x_n \rangle$ to $M$. A team $S$ is a set of agents. A team $S$ is admissible for objects of sort cn if $dom(S) = \langle x_1, \ldots, x_n \rangle$ and for $x_i$, $1 \leq i \leq n$ the sort of $x_1$ $v(i)$ is admissible for objects of sort cn.*

Each row in Table 2 is an assignment, or, when viewed from the point of view of an application, a possible description of an object (an agent). Properties of objects (agents) are represented by attributes which are variables in the formal representation. Thus, teams are directly related to the view of an object as a 'bundle' of properties.

An operation on teams is the *supplement* operation, which adds a new attribute to the objects in a team, or alternatively changes the value of an existing attribute.

**Definition 5 (Supplement of a team, [Vää07])** *If $M$ is a set, $S$ is a team with $M$ as its codomain and $F : S \to M$, $S(F/x_n)$ is the supplement team $\{s(F(s)/x_n) : s \in S\}$, where $s(a/x_n)$ is the assignment which agrees with $s$ everywhere except that it maps $x_n$ to $a$: $dom(s/x_n) = dom(s) \cup \{x_n\}$, $s(a/x_n)(x_i) = s(x_i)$ when $x_i \in dom(s) \setminus \{x_n\}$ and $s(a/x_n)(x_n) = a$.*

The supplement operation is used to model the combination of the lexical meaning of a common noun with context information and world knowledge about the referent of this noun in a given context. Let $x_n, \ldots, x_m$, $n < m$, be the attributes about which the context and world knowledge provide information. If $S$ is the team corresponding to the lexical meaning of a common noun, then $S(F/x_n)(F/x_{n+1}) \ldots (F/x_m)$ is the team resulting from adding the information aboute the attributes $x_n, \ldots, x_m$.

In Dependence Logic, formulas are interpreted with respect to sets of assignments (teams) and not w.r.t. to single assignments as in FOL. In Dynamic Dependence Logic, formulas are interpreted as relations between sets of assignments, [Gal13]. This shift makes it possible to define dependency relations between attributes. For example, functional dependence between a sequence $\boldsymbol{x}$ of variables and a variable $y$ is expressed by the atomic formula $=(\boldsymbol{x}, y)$, with the intuitive meaning 'the $\boldsymbol{x}$ totally determine $y$'. The satisfaction clause for this dependence atom is (9-a). The constancy atom $=(x)$ requires the value of the attribute $x$ to be constant in a team, (9-b).[2]

(9)
a. $\mathcal{M} \models_S =(\boldsymbol{x}, y)$ iff $\forall s, s' \in S(s(\boldsymbol{x}) = s'(\boldsymbol{x}) \to s(y) = s'(y))$
b. $\mathcal{M} \models_S =(y)$ iff $\forall s, s' \in S(s(y) = s'(y))$
c. $\mathcal{M} \models_X \exists x \phi$ iff there is a function $F : X \to \exists_M$ such that $M \models_{X[F/x]} \phi$, where $\exists_M$ is the local existential quantifier defined by $\{A \subseteq M \mid A \neq \emptyset\}$ and $S[F/x]$ is the team $\{s[a/x] \mid s \in S, a \in F(s)\}$.

The interpretation of the existential quantifier is based on the supplement operation, i.e. it either adds a new attribute to all agents in the current team, or alternatively it changes the value of an existing attribute. Thus, the existential quantifier is inherently dynamic in the sense that it changes the current team w.r.t. which it is interpreted (see [Gal13] for details on a dynamic interpretation of Dependence Logic).

Using the dependence formula $= (\boldsymbol{x}, y)$, it is possible to express dependencies between properties like 'Age' and 'Price' for stamps and 'Ripeness' and 'Sound' for melons.

(10)
a. $=(age, price)$
b. $=(ripeness, sound)$

Both examples in (10) are not quite correct because they do not take into consideration that for example (10-a) holds for stamps but not for other artefacts or human beings. Second, the value of the price depends in general not only on

[2] For formulas that do not contain a dependence atom, one has: $\mathcal{M} \models_S \phi$ iff for all $s \in S : \mathcal{M} \models_s \phi$, where $\models_s$ is the usual Tarskian satisfaction relation.

its age but also on other factors like availability or demand. These shortcomings can be remedied as follows.

(11) a. $x_{sort} = stamp \rightarrow =(age, price)$
b. $x_{sort} = stamp \rightarrow =(age, availability, demand, \ldots, price)$

A team represents the set of possibilities of a language user in the following sense: $g \in S$ if and only if the language user believes $g$ to be a possible (and complete) description of the object. As noted in [GV13], moving from assignments to teams (or sets of assignments), makes it possible to assign to each formula $\phi$ and model $\mathcal{M}$ the family of teams $\mathcal{S} = \{S \mid \mathcal{M} \models_S \phi\}$. As a consequence, formulas can be interpreted as *conditions over belief sets*. Knowledge of the value of a property in the sense this property is assigned the same value in all information states can be expressed by a constancy atom $=(x)$. In Table 2 above, this holds for the attributes 'sort', 'form' and 'origin'.

**Definition 6 (Information state w.r.t. to an object)** *Given a decompositional formula $\phi$ representing the beliefs of a language user about an object $o \in O$, his epistemic uncertainty (or his set of possibilities) w.r.t. to $o$ is given by the family of teams $\mathcal{S}$ of teams satisfying $\phi$, i.e. $\mathcal{S} = \{S \mid \mathcal{M} \models_S$.*

Note that information states are defined w.r.t. the domain $O$ of objects. The domain $E$ of events plays no role. Rather, this domain functions as a state transformer: elements of this domain trigger changes in information states.

### 3.3 Ranking functions

So far, the information state about an object of a language user is flat in the sense that all possible worlds in this information state are taken as equally plausible. However, a language user also has expectancies about (i) the values of properties about which he so far doesn't have any information and (ii) sorts of events in which an object of the given sort is most plausibly involved. Such expectancies are defined in terms of $\kappa$-rankings.

**Definition 7 ($\kappa$-ranking function; [GP92])** *A ranking is a function $\kappa : \Omega \rightarrow \mathbb{N}^*$ with $\Omega$ a non-empty set such that $\kappa(\omega) = 0$ for at least one $\omega \in \Omega$ and $\mathbb{N}^* = \mathbb{N} \cup \{\infty\}$.*

In the present context, $\Omega$ is either a set of possible words or the domain $E$. The numbers can be thought of as denoting degrees of surprise [Hal05, 43]. In terms of plausibility or expectancy, the value 0 means 'most plausible' or 'most expected'. The value $\infty$ means 'impossible' or 'so surprising as to be impossible'. $\kappa$-rankings can also be used to assign degrees of plausibility to formulas.

(12) a. $\kappa(\phi) = min_{\mathcal{M} \models_w \phi}\{\kappa(w)\}$
b. $\kappa(\psi|\phi) = \kappa(\psi \wedge \phi) - \kappa(\phi)$
c. $\kappa(\psi \wedge \phi) = \kappa(\psi|\phi) + \kappa(\phi)$

In terms of $\psi|\phi$, defeasible conditionals (or defaults) can be defined [GP92]. The inequality $\kappa(\psi|\phi) > \delta$ means that given $\phi$ it would be surprising by at least $\delta + 1$ ranks to find $\neg\phi$. As shown in [GP92], this inequality is equivalent to $\kappa(\psi \wedge \phi) + \delta < \kappa(\neg\psi \wedge \phi)$.

(13) $\quad \phi \xrightarrow{\delta} \psi$ iff $\kappa(\psi \wedge \phi) + \delta < \kappa(\neg\psi \wedge \phi)$.

$\phi \xrightarrow{\delta} \psi$ means 'Typically, if $\phi$ then expect $\psi$ with strength $\delta$'. A ranking function $\kappa$ is admissible with respect to a set $\Delta$ of defeasible conditionals if (14) holds.

(14) $\quad \kappa(\phi_i \wedge \psi_i) + \delta_i < \kappa(\phi_i \wedge \neg\psi_i)$ for all $\phi \xrightarrow{\delta} \psi \in \Delta$.

### 3.4 Rankings on information states

In a first step, the set of teams $W$ satisfying a decompositional formula $\phi$ is ranked.

**Definition 8 (Ranking on information states)** *A ranking on an information state corresponding to a decompositional formula $\phi$ is a ranking function $\kappa : W \to \mathbb{N}^*$ s.t. $\kappa^{-1}(0) \subseteq [[\phi]]$ iff $\mathcal{M} \models_w = (\phi)$ for all $w \in W$. This condition expresses the requirement that a language user knows the value of a property if it is constant in all teams satisfying $\phi$.*

The ranking function $\kappa$ can naturally be interpreted as characterizing the degree to which a language user is willing (i) to predict possible continuations of a sentences with respect to properties of objects and (ii) to accept alternative descriptions which are not in accordance with his current information about the object. For example, in the case of a melon or an orange, the most plausible values for the attribute 'Taste' is 'fruity', whereas 'salty' will most likely get the value $\infty$ because it is deemed to be impossible. One has $\kappa(\phi) < \kappa(\psi)$ if $\phi$ is more expected than $\psi$. If a language user only knows that the object is of sort $P_{cn}$, only $\phi_\sigma$ satisfies the condition $\kappa^{-1} \subseteq [[\phi_{sigma}]]$.

### 3.5 Rankings of information states on events

A relation between $W$ and $E$ is defined in terms of an *event ordering*. Each world $w$ has assiciated with it an event ordering $\mu(w)$ that determines the plausibility of event occurrences at that world.[3]

**Definition 9 (Event ordering, [Bou98])** *An event ordering is a mapping $\mu : W \to (E \to N^*)$ that maps each $w \in W$ to a $\kappa$-ranking $E \to N^*$ on the domain of events E. Instead of $\mu(w)$, we will write $\kappa_w$. It is required that $\kappa_w(e) = 0$ for some event $e \in E$, i.e. there is at least one most plausible event to occur in a world $w$. If $\kappa_w(e) = \infty$, this means that an occurrence of $e$ at $w$ is taken to be impossible. In addition we require $\kappa_w(e) = \kappa_w(e')$ for two events $e, e'$ belonging*

[3] Intuitively, $\kappa_w(e)$ captures the plausibility of the occurrence of event $e$ at $w$.

*to the same sort $P_v$, i.e. events of the same sort are assigned identical plausibility for a given $w$.*

For example, if $W$ is a family of teams of sort 'duck', events of sort 'swim' will be assigned the value 0. By contrast, if the sort is 'deer', events of sort 'jump' are most plausible and hence get value 0. For human beings, the set of most plausible events is in general rather large due to the fact that they can be correlated to a large number of different sorts of events (see [vEvSZB10] for details).

Since $W$ represents information about objects, the mapping $\mu$ establishes a relation between the domain $O$ and the domain $E$. The cognitive significance of this mapping is the following. Given an information state $w$, a language user uses $\kappa_w$ to defeasibly infer the most plausible events that are likely to occur with an information state of this sort and, in an additional step, expects particular verbs (or verbs stems) to occur farther down the sentence which denote events of those sorts.

### 3.6 Rankings of information states w.r.t. events

The mapping $\mu$ only captures the expectance of the occurrence of an event given objects of a particular sort. Next we define an analogous mapping that determines the expectancy of a particular sort of object, given information about an event of some sort.

**Definition 10 (Information state ranking for events)** *An information state ranking for events is a mapping $\mu^* : E \to (W \to \mathbb{N}^*)$ that is defined by $\mu^*(e)(w) = \mu(w)(e)$.*

The cognitive significance of this mapping is similar to that of $\mu$. If a verb is encountered denoting events of type $\sigma$, a language users uses this mapping to predict the most plausible sorts of objects to fill in a role in the event.

### 3.7 Event outcome ranking

In a final step, we define the relation between an event and its possible outcomes. This relation depends on an input state and maps an event $e \in E$ and an information state $w \in W$ to a ranking function on $W$.

**Definition 11 (Event outcome ranking)** *An event outcome ranking is a mapping $\tau : E \to (W \to (W \to \mathbb{N}^*))$ that assigns to an event $e \in E$ and an (input) information state $w$ a ranking function on the set of information states. It is required that $\forall e, e' \in P_v : \tau(e)(w) = \tau(e')(w)$ hold, i.e. events of the same sort have the same outcome ranking functions relative to a given information state $w$. Since $\tau(e)(w)$ is a ranking function, one must have $\tau(e)(w)(w') = 0$ for at least one event $w'$ so that one outcome of $e$ is most plausible.*

Intuitively, $\tau(e)(w)(w')$ describes the plausibility that the world $w'$ results when event $e$ occurs in $w$ (Boutilier 1998:292). For example, an event denoted by 'ripen' results in a state in which the object that undergoes the change, say a melon, is ripe.[4] The cognitive significance of $\tau$ is the following. If a language user knows the sort of the event, say after having processed the predicate, he can defeasibly infer possible outcomes.

## 4 Applying the formalism to the data from section 1

When processing a sentence, a language users knows that his current information state will be changed to a new one. Using his world knowledge, he also knows that this sentence either describes a change in the world or the persistence of a property of an object. In the former case the event described can either be deterministic or non-deterministic and the sentence can describe a relation between two properties over time that are linked by a dependency relation.

The cognitive significance of ranking functions and dependeny relations is grounded in the fact that they allow a language user to anticipate as much information as possible about the potential output information state that results from processing the next upcoming sentence. Using the mappings $\kappa$, $\mu$, $\mu^*$ and $\tau$, he can already calculate the plausibility of a transition $w \xrightarrow{e} v$ as follows (Boutilier 1998:292).[5]

(15) $$\kappa(w \xrightarrow{e} v) = \tau(e)(w)(v) + \mu(w)(e) + \kappa(w).$$

According to (15), the plausibility of a transition $w \xrightarrow{e} v$ depends on the plausibility of $w$, the degree to which an event $e$ is expected to occur in $w$ and the degree to which event $e$ can bring about an outcome $v$ given input $w$. Given a condition $\phi$ that has to hold in the output state $v$, the set of possible $\phi$-transitions is defined by (16) (Boutilier 1998:293).

(16) $$Tr(\phi) = \{w \xrightarrow{e} v \,|\, v \models \phi \wedge \kappa(w \xrightarrow{e} v) \neq \infty\}.$$

The most plausible transitions resulting in an outcome state satisfying $\phi$ are (17).

(17) $$\mathrm{mpt}(\phi) = \{v \,|\, w \xrightarrow{e} v \in min(Tr(\phi))\}.$$

In our application to natural language, the interpretation of a sentence need not involve all three mappings. For example, an N400 effect can be triggered both for stative sentences like 'The melon sounds muffled' and for sentences describing a change in the world like 'John squeezed an orange'. Thus, this effect is independent of the question of (i) whether the sentence describes a change in

[4] 'The melon ripened' implies that the melon was ripe at the end of the event since 'ripen' is a so-called degree achievement.

[5] As noted by Boutilier (1998:292), this formula is the qualitative analogue of the probabilistic equation $Pr(w \xrightarrow{e} v) = Pr(v|w,e) \cdot Pr(e|w) \cdot Pr(w)$.

the world or the persistence of the value of a property and (ii) if the event is deterministic or not and if dependency relations are involved.

For the analysis of sentences, the induced rankings on formulas are used to defeasibly infer additional information from a team representing the information provided by the lexical meaning, the context and world knowledge. Defeasible inferences are formulated in terms of *defeasble conditionals*. We follow [GP92], who define a consequence relation on a set $\Delta$ of defeasible conditionals and a distinguished $\kappa$-ranking $\kappa^+$. This ranking is defined as a ranking function that is minimal in the sense that any other admissible ranking function must be assigned a higher ranking to at least one world and a lower ranking to none.[6] As a consequence, $\kappa^+$ assigns to each world the lowest possible rank permitted by the admissible constraint. In our approach, the information in the antecedent is information about an input information state whereas the consequent contains information that is added to this information state and that is therefore part of the output information state. Formally, this distinction between the input information and the output information state shows up in the fact that in the consequent a formula of the form $\exists x.\phi$ is used, expressing that the current information state is changed (or supplemented). In the case of $\mu$ and $\mu^*$, defeasible inferences only use information about the sort of an event (an object) in the antecedent and information about an object (an event) in the consequent. Since the outcome mapping $\tau$ involves a relation between events and informations states, not only sortal information, but also information about other attributes is used.

**Definition 12 (Plausible inference, [GP92])** *$\sigma$ is a plausible conclusion of $\phi$ relative to a set $\Delta$ of defeasible conditionals, written $\phi \mathrel{|\!\!\!\sim^{\delta}} \sigma$, iff $\kappa^+(\phi \wedge \sigma) < \kappa^+(\phi \wedge \neg\sigma)$.*

**Simple sentences and the N400.** For simple sentences like ‘John squeezed an orange’, only the mappings $\mu$ and $\mu^*$ are important. Outcomes play no role because only the expectancy relations between sorts are involved. In (18), two examples of plausible inferences are given (somewhat simplifying, it is assumed the an object frame has an ‘event’-attribute).

(18) a. $x_{sort} = squeeze \mathrel{|\!\!\!\sim^{0}} \exists x(x = x_{theme} \wedge x = orange)$.
b. $x_{sort} = orange \mathrel{|\!\!\!\sim^{0}} \exists x(x = x_{event} \wedge x = squeeze \vee x = buy)$.

Example (18-b) is used to augment the current state with the information that the eventuality is of sort ‘squeeze’ or of sort ‘buy’. If $\delta > 0$ holds, this means that a language user is more reluctant to draw the plausible inference. However, in the present context it is assumed that a language user only uses plausible inferences where $\delta = 0$. In the consequent, the existential quantifier is used, in order to capture the dynamic character of this defeasible inference since a new attribute, here $x_{event}$ has to be introduced.

**The *the* … *the*-construction and the inferential user of PBVs.** In contrast to simple sentences like ‘John squeezed an orange’, which can be ana-

[6] [GP92] show that any consistent set $\Delta$ has a minimal ranking.

lyzed in terms of only using $\kappa$ and $\mu$, both the *the* ... *the*-construction and the inferential use of PBVs involve in addition the outcome mapping $\tau$. This is a direct consequence of the fact that they involve dependencies of (the values of) properties over time.

**The *the* ... *the*-construction.** Consider again example (6-a), repeated here as (19).

(19) The older a stamp, the more expensive it is.

The price of a stamp is in general not only dependent on its age but also on other factors such as availibility and demand. Recall that in Dependence Logic this dependence is expressed by $=(\boldsymbol{x}, \mathit{price})$, where $\boldsymbol{x}$ is a sequence of variables (attributes) containing 'age'. Such a functional dependence is a necessary condition for the truth of a *the* ... *the*-construction. In addition, a stamp can get older without becoming more expensive at the same time. Thus, one only has 'Typically (normally), a stamp gets more expensive if it gets older'. Therefore, an event of sort 'ageing' (or 'getting older') for a stamp can have at least two different outcomes. In one output only the age of the stamp has increased and in a second output both its age and its price have increased (relative to the input state). As a consequence, events of ageing for stamps are non-deterministic. Since the *the* ... *the*-construction involves the comparative construction *the* ... *the*, it is necessary to not only consider single transitions as defined in (15), but histories of such transitions.

(20) $w \xrightarrow{e^n} v$ iff there are $u_0, \ldots, u_n$ s.t. $w = u_0$ and $v = u_n$ and for each $(u_i, u_{i+1})$ with $0 \leq i < n$ there is an $e$ s.t. $u_i \xrightarrow{e} u_{i+1}$.

For $w \xrightarrow{e^n} v$, the plausibility relation on histories is calculated in terms of the plausibility of its (atomic) transitions. This plausibility expresses the degree to which a language user thinks that this history might occur (or has been occurred, using an abductive argument).

Each history represents a possible evolution of how an outcome $\phi$ can be brought about by a sequence of events $e = e_1 \ldots e_n = e^n$. If the sort of the events $e_i$ is restricted to events of sort 'ageing', all histories have an outcome in which the object undergoing the change is older than in the input state. The output states can differ w.r.t. other properties, like 'Price' for example, that can also be changed by an event of sort 'ageing'.

The output states $u_i$, $1 \leq i \leq n$, of the (atomic) transitions differ in the value assigned to the outcome mapping $\tau$ (They do not differ w.r.t. $\kappa$ and $\mu$ because an event of ageing leaves these rankings unchanged). Assuming $\tau(e)(w)(v) = 0$, just in case $v$ satisfies both the condition that the value of 'Age' has increased and that the value of 'Price' has increased, the most plausible histories involving a sequence of ageing events for a stamp are those in which the stamp gets both older and more expensive.

For the outcome mapping $\tau$, the basic form of a defeasible conditional has the form (21-a). Since $\tau$ maps an event and an input state to a ranking function $\kappa$

on $W$, the antecedent refers both to information about an event and information about an object represented in $W$. For (19), one gets (21-b).

(21) a. $x_{e_{sort}} = \sigma_e \wedge x_{o_{sort}} = \sigma_o \wedge x_{o_{attr}} = \beta \overset{\delta}{\mid\!\sim} x_{o_{attr}} \neq \beta$.

b. $x_{e_{sort}} = ageing \wedge x_{o_{sort}} = stamp \wedge x_{o_{age}} = \alpha \wedge x_{o_{price}} = \beta \overset{\delta}{\mid\!\sim} \exists x(x = x_{o_{age}} \wedge x > \alpha) \wedge \exists x(x = x_{o_{price}} \wedge x > \beta)$

**Inferential use of PBVs.** On its inferential use, the interpretation of a PBV involves a change. We will argue that the interpretation process is similar to an abductive argument (see Boutilier 1998) involving three steps. Consider the example 'The melon sounds ripe'. First, there is an observation (perception): the melons emits a particular sound that is classified as 'muffled'. Second, an explanation for this particular sound value is given by postulating some (most) plausible event or events that could have brought about the observed change in the property expressed by the verb ('sound' in this case). Besides a ripening event, the sound of the melon could have been manipulated mechanically. But the former event is assumed to be more plausible, say due to experience and general world knowledge. Finally, the outcomes of this event are calculated. In this case one gets that the melon is ripe. The defeasible element is the postulation of a (most) plausible event. In the case of PBVs, this is an event related to the property expressed by the predicative complement, e.g. a ripening in the case of 'The melon sounds ripe', where the predicative complement is 'ripe'.

Similar to the *the* ... *the*-construction, there are two constraints that must be satisfied. First, there must be a functional dependency between the two properties. For example, the value of the 'Sound' property must be determined by the value of the 'Ripeness' property. Second, this condition need only hold in the normal or typical case. Consider '*The melon sounds oval'. In this case there is no functional dependence between the value of the property 'Sound' and the property 'Form'. In a team of sort 'melon', the value of the 'Form' property can arbitrarily vary while the 'Sound' property remains constant, say 'muffled'. For 'The melon sounds muffled', the information in the input information state is (22-a). The first defeasible inference is based on the mapping $\mu$, (22-b). The second step involves the non-defeasible inference that an event denoted by the verb 'ripe' brings about a state in which the object undergoing the ripening is ripe at the end of the event (22-c). When taken together, one gets (22-d).

(22) a. $sort_{o_{sound}} = muffled \wedge x_{o_{sort}} = melon$.

b. $sort_{o_{sound}} = muffled \wedge x_{o_{sort}} = melon \overset{0}{\mid\!\sim} \exists x(x = x_{e_{sort}} \wedge x = ripen)$.

c. $x_{e_{sort}} = ripen \wedge x_{o_{sort}} = melon \vdash x_{o_{ripeness}} = ripe$.

d. $sort_{o_{sound}} = muffled \wedge x_{o_{sort}} = melon \wedge x_{e_{sort}} = ripen \overset{0}{\mid\!\sim} \exists x(x = x_{o_{ripeness}} \wedge x = ripe)$.

## 5 Summary

In this paper we developed a dynamic semantic theory which makes it possible to express both dependency relations between properties of objects and expectancies between nominal and verbal concepts. The theory is based on a decompositional analysis of common nouns in which they are interpreted as 'bundles' of properties, similarly to the way objects are represented in databases theories. The ranking functions are used to draw defeasible inferences provided words.

Needless to say, the theory has to be worked out in greater formal detail: (i) The relation between Dependence Logic and $\kappa$-rankings must be further explored. E.g., is it possible to define ranking functions directly in Dependence Logic?; (ii) The dynamic component must be made more explicit. In particular, how are information states for various objects modeled and how is it possible to explicitly talk about changes?; and (iii) How are the rankings empirically determined? Possible approaches are strategies from n-gram models and techniques used in neuroscience based on the concept of cloze probability.

## References


[BH11] Giosuè Baggio and Peter Hagoort. The balance between memory and unification in semantics: A dynamic account of the N400. *Language and Cognitive Processes*, 26(9):1338–1367, May 2011.

[Bou98] Craig Boutilier. A unified model of qualitative belief change: A dynamical systems perspective. *Artificial Intelligence*, 98(1-2):281–316, 1998.

[FK99] Kara D. Federmeier and Marta Kutas. A rose by any other name: Long-term memory structure and sentence processing. *Journal of Memory and Language*, 41:469, 1999.

[Gal13] P. Galliani. Transition semantics: the dynamics of dependence logic. *Synthese*, 2013. Published online: 11 August 2013.

[GP92] Moisés Goldszmidt and Judea Pearl. Rank-based systems: A simple approach to belief revision, belief update, and reasoning about evidence and actions. In Bernhard Nebel, Charles Rich, and William R. Swartout, editors, *Proceedings of the 3rd International Conference on Principles of Knowledge Representation and Reasoning (KR'92)*, pages 661–672. Morgan Kaufmann, 1992.

[GV13] Pietro Galliani and Jouko A. Väänänen. On dependence logic. *CoRR*, abs/1305.5948, 2013.

[Hal05] Joseph Y. Halpern. *Reasoning about uncertainty.* MIT Press, 2005.

[Löb14] Sebastian Löbner. Evidence for frames from human language. In Thomas Gamerschlag, Doris Gerland, Rainer Osswald, and Wiebke Petersen, editors, *Frames and Concept Types*, volume 94 of *Studies in Linguistics and Philosophy*, pages 23–67. Springer International Publishing, 2014.

[Vää07] J. Väänänen. *Dependence Logic.* CUP, Cambrige, 2007.

[vEvSZB10] Michiel van Elk, Hein T. van Schie, Rolf A. Zwaan, and Harold Bekkering. The functional role of motor activation in language processing: Motor cortical oscillations support lexical-semantic retrieval. *NeuroImage*, 50(2):665–677, 2010.