

# How Factuality Determines Sentiment Inferences

Manfred Klenner and Simon Clemenide

Computational Linguistics  
University of Zurich, Switzerland  
{klenner|siclemat}@cl.uzh.ch

## Abstract

In a complex sentence comprised of one or more subclauses, the overt or hidden attitudes between the various entities depend on the factuality projection of the verbs, their polar effects, and the modality and affirmative status (negated or not) of the clauses. If factuality is given, some referents might even be considered to benefit or to suffer from the (effects of the) described situation, independently of their relations to the other referents. An interesting question is, how the reader evaluates all this from his/her perspective. We introduce an approach based on Description Logics that integrates these various perspectives into a joint model.

## 1 Introduction

Sentences can express a positive or negative relationship between people, organizations, and nations etc. For instance, in the sentence “the EU supports Greece”, a positive attitude of the EU towards Greece is expressed. At the same time, a positive effect that is meant to be true, is asserted. That is, Greece benefits from the situation described. If the reader has a negative attitude towards the beneficiary (Greece), he might regard the apparent benefactor (EU) as his opponent. However, if the sentence is embedded into a non-factive verb like “to pretend” (“The EU pretends to support Greece”), neither the positive relationship between the referents nor the positive effect on Greece hold any longer. Instead, the matrix verb “to pretend” casts a negative effect on the EU. If the reader adheres to this common sense verb connotation, he will adopt the negative attitude towards the EU. Furthermore, if some actor criticizes that the EU supports Greece, factuality of the embedded clause is given (compared

to “pretend”). Thus, the positive effect on Greece still takes place, but now there is a negative attitude of this actor of the matrix clause towards both referents of the complement clause. Finally, if an actor criticizes that the EU does *not* support Greece, his attitude towards Greece is positive (but negative towards the EU).

Given a text, we would like to be able to answer the following questions: What is good or bad *for* the entities mentioned in the text? What is good or bad *of* these entities? What are the attitudes of the entities towards each other? And last but not least, what follows from the reader’s stance, i.e. his prior attitudes towards some entities?

A user of our system then could mine texts for proponents and opponents of his, in the sense that entities that do things (or like others that) he likes are proponents, and entities that act in the opposite way (or like others he dislikes) are opponents.

In contrast to existing work (e.g. Deng and Wiebe (2015)), we stress the point that verb signatures in the sense of Karttunen (2012) that capture (non-)factuality information regarding complement clauses need to be taken into account in order to properly draw such inferences. We focus on complex sentences where a matrix verb restricts its subclauses with respect to factuality depending on its affirmative status (i.e. whether the matrix clause is affirmative or negated). The interplay of (non-)factuality with negation, the various polar restrictions projected by the verbs, and the aforementioned relational layer give rise to a complex model.

We have implemented a joint model with Description Logics (DL), namely OWL (Horrocks and Patel-Schneider, 2011) and SWRL (Horrocks and Patel-Schneider, 2004). The model is language-independent. However, the mapping from a sentence to input structures is mediated by a dependency parser, a predicate-argument extractor and a verb lexicon covering the polar restric-

tions – these components are language-dependent. We give English examples in this paper, although our pipeline (and the empirical evaluation) is for German. Our English example sentences were manually converted to OWL representations.

## 2 Related Work

The topic of event factuality in natural language applications is thoroughly discussed in Saurí and Pustejovsky (2009). For their FactBank annotations, they differentiate between *factual* (it is the case) and *counterfactual* (it is not the case).

The *certainty* (epistemic modality) to which factuality holds is a continuum, but according to Saurí and Pustejovsky (2009) it has often been divided into the following three-fold distinction that they also adhere to: *certain*, *probable*, and *possible*. Saurí and Pustejovsky (2009) additionally provide annotation labels for cases where the factuality is underspecified. An important trait of their approach lies in the fact that these annotations are always relative to sources mentioned in the text, typically subjects or objects of source-introducing predicates, for instance, “*said the minister*”. In our work, we focus on the identification and extraction of *certain facts* that convey polar effects, opposition or support.

A rule-based approach to sentiment inference is Neviarouskaya et al. (2009). Each verb instantiation is described from an *internal* and an *external* perspective. For example, “to admire a mafia leader” is classified as affective positive (the subject’s attitude towards the direct object) given the internal perspective while it is (as a whole) a negative judgment, externally (here the concepts introduced by the Appraisal theory are used, cf. Martin and White (2005)). However, the authors do not give any details about how they carry out rule application, and factuality does not play any role in their work.

The same is true for Reschke and Anand (2011). They capture the polarity of a verb frame instantiation as a function of the polarity of the verb’s roles. In our approach, we do not assume to know the polarity of the roles in advance, but intend to infer them contextually. In their approach, if a murderer loses something positive, then this is positive as a whole. It is hard to see how less drastic cases are to be treated. For instance, “the thief loses all his friends” – is this positive? We would say: it is negative for the thief and that the friends have a

negative attitude towards the thief.

How Description Logics can be used to identify so-called polarity conflicts was described in Klenner (2015). However, attitudes and the factuality of situations were not part of that model.

## 3 The Verb Model: Polarity Frames

The basis of our approach is a verb resource that we call polarity frames (Klenner et al., 2014; Klenner and Amsler, 2016). The current lexicon is comprised of 330 German verbs that instantiate 690 polarity frames. A verb can have more than one polarity frame due to polysemy. We are particularly interested in those verbs that subcategorize complement clauses (78 verbs), since they are crucial for complex inferences.

For each argument of a polarity frame (agent, patient, theme, etc.), we specify whether it casts a polar effect on its argument filler. For instance, the patient argument of “to help” receives a positive effect. We distinguish between polar roles that indicate that something is good/bad *of* or *for* someone. The agent role is an *of-role* – it is good *of* A to help B. The patient role (depending on the verb also theme or recipient) is a *for-role*, i.e. it is good *for* B if A helps her.

Given the verb “to help”, there are at least two polarity frames, the transitive one (“A helps B”) and the one with an embedded (infinitival) subclause (“A helps to XCOMP”). In the first frame, both argument fillers receive a positive effect. The agent is a positive *of-role*, which we call the *posof* role. Accordingly, the patient is a *posfor* role. Both roles are generalizations of the traditional semantic roles.

In the second frame (“A helps to XCOMP”), the agent again is the bearer of the *posof* role. But now it is XCOMP that receives a positive effect, i.e. it is good for the situation denoted by XCOMP to receive help. Thus, not only entities but also situations are affected by the polarity that a verb casts on its arguments. In order to distinguish roles for situations from roles for entities, we call the roles for positively and negatively affected subclauses *poscl* and *negcl*, respectively. This nomenclature (*posof*, *posfor*, *poscl*) eases the development of general inference rules over entities and situations.

### 3.1 Verb Signatures

Verbs that subcategorize a clausal complement are further specified for factuality of the clausal com-

Label	Explanation	Matrix Verb
F	factual in any case	to regret
NF	non-factual in any case	to hope
AF	factual if affirmative	to force
ANF	non-factual if affirmative	to forget
NaF	factual if non-affirmative	to forget
NaNF	non-factual if non-affirmative	to manage
NaO	true or false if non-affirmative	to help

Table 1: (Non-)Factuality of subclauses

plement. Factuality means that the situation described in the subclause is meant (by the author) to be true (to hold). We follow the work of Karttunen (2012), who distinguishes factive, non-factive and implicative verbs. Factuality of the subclause depends on the matrix verb’s signature and the presence or absence of negation in the matrix clause.

Table 1 summarizes the signatures of example matrix verbs and introduces our short labels (e.g. AF). Factive verbs, such as “to regret”, cast factuality on their subclause, whether the main clause is negated or not. If A regrets that COMP, then COMP is true in the sense that the speaker believes (or a least asserts) COMP to be true. The same holds for “A does NOT regret that COMP” (factuality here is constant under negation, thus factuality is a presupposition of factive verbs). Subclauses of non-factive verbs, on the other hand, are never meant to be factual (e.g. “to pretend”, “to hope”).

Then, there are verbs called implicatives that cast a mixture of factuality and non-factuality. Two-way implicatives, like “to forget to”, have non-factual subclauses in an affirmative use, but factual subclauses if negated. One-way implicatives only give rise to factuality in either the affirmative (like “to force”) or negated matrix verb contexts (like “to refuse”). For instance, if A forces B to lie, B lies. If A does not force B to lie, then B might lie as well, we just cannot tell.

Non-factuality blocks some, but not all inferences. In “A hopes that B wins”, the subclause is non-factual, so B does not receive a positive effect (he is not a beneficiary): this inference is blocked. However, the attitude of the *of-role* of the (factual) matrix sentence (A) towards the *for-role* of the (non-factual) embedded verb holds (a positive relationship): it is not blocked. Relationship inference *within* a non-factual clause, however, is blocked, e.g. if A hopes that B loves C, the inference that B has a positive attitude towards C is blocked.

Verb	of-role	for-role	cl-role	aff	neg
criticize	of	n/a	negcl	AF	NaF
approve	of	n/a	poscl	AF	NaF
help	posof	n/a	poscl	AF	NaO
help	posof	posfor	n/a	n/a	n/a
survive	n/a	posfor	n/a	n/a	n/a

Table 2: Polarity frames

Table 2 shows the polarity frames of some verbs. The polar roles *poscl* and *negcl* stand for positive and negative effects of the verb on its subclause (*cl-role*), respectively, while *of* indicates a neutral effect. The last two columns relate to the verb signatures as introduced in Table 1, the second last column reports the restriction whether the matrix verb is aff(irmative) and the last column whether it is neg(ated). For example, the subclause of “help” (row 3) is factual if the “help” sentence is affirmative (AF), but its truth value is unspecified (NaO) if negated.

## 4 Preprocessing Pipeline

Our polarity frames provide a mapping from grammatical roles to our generalized set of semantic roles, which we call the *polar semantic roles* of a verb. For instance, the subject of “to survive” is mapped to a *posfor* role while the subject of “to cheat” realizes a *negof* role. In order to provide a proper mapping, we have to identify these grammatical roles given a dependency parse. Among others, passive voice, but also implicit arguments given control or raising verbs raise the need to reconstruct the real fillers of the grammatical roles of the verbs from the surface structure of the dependency parse. Also coreference needs to be coped with.

We have implemented a rule-based polar semantic role labeler. Extraction rules were automatically learned from treebank parses and the corresponding, manually annotated verb frame instantiations. Given a parse tree and a gold standard annotation of the underlying verb frames, paths between the verbs and the heads of their grammatical roles can be derived and saved as extraction patterns. Given proper verb frame instances, each filler of a grammatical role is mapped to a polar role according to the polarity frame of the verb.

Clearly, there is a great number of syntactic variations that need to be accounted for. However, 80 to 100 well-chosen correct sentences might already cover the most frequent cases of syntactic variation (cf. Klenner and Amsler (2016)).

Effect	Attitude	Reader
beneficiary	pro	MyOpponent
benefactor	con	MyProponent
victim		SympathyEntity
villain		NonSympathyEntity

Table 3: Projections: Concepts and Properties

## 5 The Overall Model

We strive to combine three different perspectives in a joint model. Firstly, there is the question of who actually profits (or has a disadvantage) from the described situation. We call this the layer of *effect projection*. Secondly, there is the relational level that determines the attitudes of the participants towards each other, this is called the *attitude projection*. Both are derived from the input text and represent the way the text puts the world (the *text perspective*). Thirdly, there is the perspective of the reader, the *reader projection*: what he or she takes from it. From the *text perspective*, the attitudes of the author (the author projection) sometimes are evident, but in the sentences envisaged by our approach this is normally not the case. We focus on sentences that report the view of the subject of the matrix clause (“A criticizes that ...”).

Table 3 shows the concepts and properties (relations) of these projection layers: The inference task is to instantiate them given a sentence, only (*Non*)*SympathyEntity* are specified in advance by the user (reader). The starting point of the inference process are the instantiated polarity frames derived from an input sentence, say, “the EU helps Greece”. We know from a dependency parse that “Greece” is the object of “help” and the polarity lexicon tells us that the object of “help” realizes a *posfor* role. This is the core of our lexical resource: grammatical roles are mapped to semantic roles (mainly specializations of for-roles and of-roles). The sentence is affirmative and since no modal verbs or modifiers are present it is factual. In a factual, affirmative sentence, the filler of the *posfor* role is a beneficiary. A *beneficiary* in our setting is someone who actually benefits from the situation described and must not be confused with the thematic role *beneficiary* from the literature: If the sentence would be negated, the beneficiary status of Greece no longer would hold. It would still occupy the *posfor* role, but since negated, it would no longer count as an entity that has received a positive, beneficial effect from the situation. On the contrary, it would now be a *victim*, since it is

denied help.

The properties *pro* and *con* establish the attitude projection. A *pro* relation represents a positive attitude, while *con* means a negative attitude. The filler of any *of-role* of a verb that also has a *posfor* role obviously has a positive attitude (a *pro* relation) towards the filler of the *posfor* role (here: EU *pro* Greece), provided again a factual affirmative use. If the filler of the *posfor* role moreover is an instance of *SympathyEntity* of the reader – this is given in advance, the user (modelled reader) has to specify which entities he likes or dislikes – then (among others) the filler of the *of-role* (EU) becomes an instance of the concept *MyProponent* of the reader (since the filler, EU, has, according to the sentence, a positive attitude, a *pro* relation, towards someone the reader likes, here Greece).

The attitude projection is realized with SWRL rules which refer to OWL concepts (e.g. *factual*) and A-Box representations of the sentence. They instantiate OWL properties which in turn are used by other OWL concepts to draw conclusions related to effects and reader projections.

## 6 Description Logics Model

Description Logics seem to be well suited for such intermingled inference tasks that we envisage. One must not care about the actual order the inferences are drawn, and global consistency checks help to identify and get rid of unwanted side effects. One drawback of pure Description Logics is that relational concepts are a problem. We cannot define a concept *opponent* that relates two individuals A and B, we always have to state a direction<sup>1</sup> namely that B is an opponent of A, i.e., B is an *A-opponent*, so to speak. We have chosen this possibility to define relational concepts w.r.t. the reader. We define the concepts *MyOpponent* and *MyProponent* to capture the reader’s perspective. However, we found it much more convenient to use SWRL rules (Horrocks and Patel-Schneider, 2004) instead of pure OWL concepts (Horrocks and Patel-Schneider, 2011) to define the remaining relational inference layer.

Our system was developed in the Protégé editor, which eased the semantical engineering task. HermiT (Glimm et al., 2014) was used for SWRL and OWL reasoning. In the following, we introduce the properties, instance representations, concepts,

<sup>1</sup>We could introduce a property *opponent*, but reasoning at the level of properties is limited.

of-role	the agent
posof	the filler gets a positive effects
negof	the filler gets a negative effects
for-role	the patient,recipient, beneficiary or theme
posfor	a positive for-role
negfor	a negative for-role
cl-role	the subclause
poscl	subclause filler receives a positive effect
negcl	subclause filler receives a negative effect

Table 4: Properties for verb argument roles

and SWRL rules of our model.

### 6.1 Properties

OWL properties represent two-placed relations between concepts, they have domain and range restrictions (we do not specify the concrete restrictions here). We have properties that realize the semantic roles of polarity frames. They are used to represent verb instantiations. We have a property *for-role* with subproperties *posfor* and *negfor* and a property *of-role* with *posof*, *negof* as subproperties. These are roles for entities. For situations, a general role *cl-role* denotes a non-polar subclause restriction (e.g. the verb “to remember that” casts it). *negcl* and *poscl* denote positive and negative effects that the matrix verb casts on its complement clause. These roles also have inversed roles, indicated by a preceding initial I (e.g. *I-posof*), to cope with the problem of bidirectional relational properties in Description Logics. Table 4 summarizes our role inventory.

*pro* and *con* of the attitude layer are also realized as properties. These properties are to be inferred by the system (as specified in section 7), in contrast to the verb argument properties from Table 4 which are instantiated via the dependency tree and the polarity frame lexicon.

### 6.2 Sentence Representation (A-Box)

We represent sentences and their verb instantiations in a manner that is inspired by Davidson’s approach (Davidson, 1966), i.e. verbs are referred to by a constant that represents a verbal event instantiation. Technically, mentions of entities and events are represented by their base form followed by a digit. For example, *survive-1* is an instance of a survive event, and *minister-1* represents a reference to a member of the class of ministers. Our example sentence “The minister has criticized that the EU has helped Greece to survive” is represented by the A-Box assertions from Table 5. The specifications are given in a slightly simpli-

<i>criticize-1</i> : (aff AND AF)	<i>help-1</i> : (aff AND AF)
<i>criticize-1 of-role minister-1</i>	<i>help-1 posof EU</i>
<i>criticize-1 negcl help-1</i>	<i>help-1 posfor Greece</i>
<i>survive-1</i> : affirmative	<i>help-1 poscl survive-1</i>
<i>survive-1 posfor Greece</i>	<i>criticize</i> : factual

Table 5: A-Box representation

fied Manchester syntax (Horridge et al., 2006).

*criticize-1* is an instance of both the classes *aff*(firmative) and *AF* (i.e. factual if affirmative; and, not shown here, *NaF*, i.e. factual if non-affirmative), and it has the role *negcl* with *help-1* as its filler. The concepts *affirmative* and *non-affirmative* are used to represent the affirmative or negated use of a predicate in a sentence.

### 6.3 Concept Hierarchy (T-Box)

As mentioned, we distinguish between the perspective of the reader, *MyView*, and the perspective of the text, *TextView*, see Fig.1. *TextView* tells us what the author believes to be true. One task of the reader as part of the understanding of a text is to find out what the text entails (class *Implication*) about the described situation (class *Situation*). A situation is either affirmative (class *affirmative*) or negated (class *non-affirmative*), which is known given the sentence (thus, both are primitive concepts). The whole sentence is meant to be true (if no modals are present), so the matrix clause is by definition factual (be it affirmative or non-affirmative). The factuality of an embedded situation (class *Embedded*) depends on the factuality class of the embedding situation denoted by the (embedding) verb (see Fig.1 for the subclasses of *Embedded*, e.g. *AF*). A factuality class like *AF* of a situation stems directly from the verb signatures, e.g. in Table 5, where *criticize-1* is an instance of *AF* since the verb “to criticize” bears that signature: whatever affirmative “to criticize” embeds, it is factual<sup>2</sup>. Thus, all subclasses of *Embedded* are primitive concepts (given by the verb signatures). Whether an embedded (individual) situation is factual or non-factual (its *Factuality\_Status*) depends on the factuality class of the embedding verb and whether the embedding verb is affirmative or non-affirmative: *factual* and *non-factual* are defined classes. The definition of *factual* in Manchester syntax is:

(I-cl-role some (F or (affirmative and AF) or

<sup>2</sup>Clearly, in: “A criticizes that B intends to lie”, the intention is factual, not the lying.

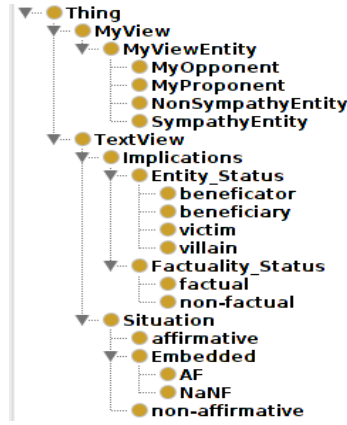


Figure 1: T-Box

(non-affirmative and NaNF))

*I-cl-role* is the inverse of *cl-role* (describing the embedding of situations). A situation is *factual* if it is embedded (*I-cl-role*) into a situation that is described by a factive verb (class *F* in Table 1), or is *affirmative* and has the signature *AF* or is *non-affirmative* and of type *NaNF*. Given this (together with the definition of *non-factual*), we are able to determine the factuality status of an embedded situation of any depth of embedding.

### 6.3.1 Effect Projection Concepts

We now turn to the *effect layer* represented by the concept *EntityStatus*. We distinguish four classes and call them programmatically *benefactor*, *beneficiary*, *villain*, and *victim*. We just give the definition of *beneficiary*. The idea behind our definition is that the beneficiary of a situation is somebody who benefits from it independently of any attitude that somebody might have towards him. So if A wins, A is the beneficiary, whether A is liked by someone or not. What must be the case is that A occupies the *posfor* role of a situation that is *factual* (not just imagined) and *affirmative* (i.e. not negated). Here is the definition of *beneficiary*:

(I-posfor some (affirmative and factual))

For convenience, we also give the predicate logic equivalent:

$$\forall x \exists y : \text{I-posfor}(x, y) \wedge \text{affirmative}(y) \wedge \text{factual}(y) \rightarrow \text{beneficiary}(x)$$

### 6.3.2 Reader Projection Concepts

The *reader layer* depends on prior information concerning the stance of the reader towards real-world entities (his prior attitudes). The user of our

system thus has to specify these kind of preferences in advance. He might state that Greece has his sympathy. This brings us to the concepts of the *MyView* class. We distinguish *SympathyEntity*, *NonSympathyEntity*, all primitive concepts. A *SympathyEntity* is either an entity that especially the reader (and maybe only he) likes (e.g. his dog) or an entity (concrete or abstract) that he, as most people from his culture, believe to be valuable (e.g. freedom). *NonSympathyEntity* is defined correspondingly.

Given the user's prior attitudes, his (non-)sympathies, and given a sentence from which the attitude projections (attitudes among the referents of the sentence) has been derived, the question is what actually makes referents opponents or proponents of the reader.

We exemplify the concept of *MyProponent* here. Trivially, any *SympathyEntity* is also an instance of *MyProponent*. However, there are more sophisticated ways to become someone who is in line with the reader's world view (*MyView*). Namely, if someone has a positive attitude (a *pro* relation) towards a *SympathyEntity* of the reader. Or, if someone is against (a *con* relation) someone the reader does not like (a *NonSympathyEntity*). Here is the definition of *MyProponent*:

(SympathyEntity or (pro some SympathyEntity) or (con some NonSympathyEntity))

The definition relies on the properties *pro* and *con*. We now turn to the part of our model which describes how to infer the referents' attitudes towards each other. The way they behave as indicated by the text determines their relationship and if at least one of the involved participants is a *SympathyEntity* or *NonSympathyEntity* of the reader, the reader projection, i.e., his opponents and proponents can be derived. If A supports B and B is a *NonSympathyEntity* of the reader, then A is an opponent of the reader (since *A con B* holds, but see the next section for the definition of these inferred properties).

## 7 Attitude Projection Rules

We use SWRL rules to specify the attitude inference layer. SWRL rules are neatly coupled with OWL concepts (T-Box) and instances (A-Box). For instance, we can refer to an instance of class *factual* by a predicate of the form *fac-*

#	Input Predicates	
1	posfor(help,GR)	negcl(criticize,help)
2	posof(help,EU)	of-role(criticize,min.)
3	poscl(help,survive)	posfor(survive,GR)
4	aff(criticize)	aff(help)
5	aff(survive)	factual(criticize)

Table 6: Input representation

$tual(?x)^3$ . Properties are referred to accordingly, e.g.  $negcl(?s,?s2)$  binds  $?s$  and  $?s2$  to any A-Box expression (in Manchester Syntax) of the form:  $someInstance1\ negcl\ someInstance2$ , e.g.  $criticize-1\ negcl\ help-1$  from Table 5. This might be somewhat intransparent to readers unfamiliar with OWL and SWRL. For convenience, we have translated Table 5 into Table 6, where A-Box expressions are mapped to a notation closer to SWRL. We also have stripped indices, e.g.  $criticize-1$  is now just  $criticize$ .

In order to introduce our scheme, we go through the example sentence S (repeated):

S: *The minister has criticized that the EU has helped Greece to survive.*

The instantiations from Table 6 are based on the polarity frames of the verbs and the dependency parse of the sentence. Since no negation is present, it holds that  $aff(criticize)$ ,  $aff(help)$ ,  $aff(survive)$  (line 4 and 5), where  $aff$  means affirmative use. The matrix clause (since no modal is present) is factual (line 5), i.e.,  $factual(criticize)$ . Note that  $posfor(help,Greece)$  just means that Greece occupies a particular polar role. Whether Greece actually gets a positive effect depends on the factuality as determined by the matrix verb and its affirmative status (and also the affirmative status of the complement verb itself).

Before reading the further outline of our rule component, the reader is invited to verify that the following inferences drawn from the example sentence S are in line with his/her intuition (i4 and i6 needs further explanation, though):

Greece as a beneficiary (i1 from Table 7) follows from the OWL definition (Greece takes the  $posfor$  role in a factual affirmative sentence).

In general, the goal is to find out whether A is for ( $pro$ ) B or whether A is against ( $con$ ) B. A verb might (directly) reveal the relation between

<sup>3</sup>We follow the SWRL notation to indicate variables by a leading question mark.

#	Inference	Rule
i1	beneficiary(Greece)	OWL def.
i2	pro(EU,Greece)	r1
i3	con(minister,EU)	r2
i4	disapprove(minister,survive)	r3
i5	con(minister,Greece)	r4
i6	con(EU,minister)	r5

Table 7: Inferences

the participants within the same clause: if A helps B, then A is  $pro$  B. If A criticizes B, then A is  $con$  B (at least in a certain – the given – context, not necessarily in a fundamental, irreconcilable way). Provided, of course, the situation is factual and affirmative.

```
r1 aff(?s), posof(?s, ?x), factual(?s),
    posfor(?s, ?y) -> pro(?x, ?y)
```

Rule r1 states: An actor  $?x$  (the  $posof$  role, in general, any of-role) is  $pro$   $?y$  if in a single factual, affirmative sentence  $?s$ ,  $?y$  is the filler of the  $posfor$  role (i2 from Table 7):  $pro(EU, Greece)$ .

If a sentence  $?s$  embeds a sentence  $?s2$ , then rules like the following are in charge:

```
r2 factual(?s), aff(?s), negcl(?s, ?s2),
    of_role(?s, ?x), of_role(?s2, ?y)
    -> con(?x, ?y)
```

According to r2, an affirmative and factual matrix clause  $?s$  that embeds an affirmative subclause  $?s2$  (the factuality of  $?s2$  is irrelevant) bearing a negative effect ( $negcl$ ) gives rise to a  $con$  relation between the  $of\_role$  of the matrix clause and the  $of\_role$  of the subclause (see i3 from Table 7):  $con(minister, EU)$ .

More complicated scenarios arise in the case of multiple embeddings. According to Table 2, “to criticize” has a  $negcl$  role while “to help” has a  $poscl$  role. If A criticizes that B helps C to D (D=survive), then, obviously, A disapproves D. That is, a  $negcl$  on a  $poscl$  gives  $disapprove$ , see rule r3.

```
r3 aff(?s), factual(?s), negcl(?s, ?s2),
    aff(?s2), of_role(?s, ?x), poscl(?s2, ?s3)
    -> disapprove(?x, ?s3)
```

The matrix clause must be factual: if A (just) *might* criticize that COMP, nothing can be inferred about A’s (dis-)approval regarding COMP (and COMP of COMP). Rule r3 triggers and produces i4 from Table 7:  $disapprove(minister, survive)$ .

The next rule describes how disapprove propagates to a  $con$  relation (factuality is irrelevant).

r4  $\text{aff}(\text{?s}), \text{posfor}(\text{?s}, \text{?y}), \text{disapproves}(\text{?x}, \text{?s})$   
 $\rightarrow \text{con}(\text{?x}, \text{?y})$

If someone disapproves an affirmative situation that is positive (*posfor*) for someone, then he is against this person. Rule r4 produces i5 from Table 7: *con(minister,Greece)*.

One could also think of rules like the following:

r5  $\text{pro}(\text{?x}, \text{?z}), \text{con}(\text{?y}, \text{?z}) \rightarrow \text{con}(\text{?x}, \text{?y})$

If A is *pro* B and C is *con* B then we might be allowed to guess that A is *con* C. In our example it follows that EU is *con* minister, see i6 from Table 7. Note that these transitively given *pro* and *con* relations are only safe if they stem from the same sentence. It is not true *in general* that I am against someone who dislikes a person I like. If (rule r5) A admires B while C finds B boring, A and C are opponents, but only conditional on B, so to speak. In general, pros and cons can only deliver situation-specific attitudes.

Now that we have seen examples of the effect projection (*beneficiary(Greece)*), the attitude projection (e.g. *con(minister,EU)*) let us end with an example of the reader projection. If the reader is skeptical about the EU (these days), i.e., the EU is a *NonSympathyEntity* of his, then *minister* becomes an instance of *MyProponent* (via the definition of *MyProponent* and the derived attitude *con(minister,EU)*).

The author projection also can be plugged in easily. Take the sentence “The minister criticizes the ridiculous initiative”. We only have to derive *con(author,initiative)* from the use of “ridiculous” and we can exploit the full capacity of our reasoning scheme, e.g. we could derive *pro(author,minister)*.

## 8 Empirical Evaluation

Our inference rules were tuned on the basis of 80 constructed development sentences (Dev80) that concisely capture our modelled phenomena. They combine verbs from our lexicon in sentences that are comprised of subclause embeddings up to three levels. Affirmative and negated use of these verbs are combined with (non-)factuality at each level of embedding. This was meant to base our model on an increased generative complexity of natural language – even if such sentences are rare in real texts. Our goal was to model competence and at same time make it applicable. The sample sentence S from the last section is an example of such a constructed sentence. For each sentence,

Relations	A	B	Gold	System
benefactor	2	2	4	5
beneficiary	10	5	7	16
victim	35	40	42	52
villain	4	5	6	11
con	68	50	68	67
pro	35	23	29	37
total	154	125	156	188

Table 8: Statistics for Test80: Annotators A and B, the adjudicated gold standard G, and the system output (setting I)

we manually instantiated the polarity frames, i.e., we identified the polarity frame and the fillers of the grammatical roles. It was the tuning of the rule component we were after, not the impact of the preprocessing pipeline (extraction from the dependency trees) on the overall performance. The final performance of our system on Dev80 was: precision 83.89% and recall 93.72%.

The final test corpus (Test80) contains 80 unseen sentences drawn from the German newspaper treebank TüBa-D/Z (Telljohann et al., 2009). About 10% of its 95,000 sentences contain a verb that is modelled in our lexicon. In about 5,000 sentences our extraction component triggers. 540 cases show subclause embedding. In 46 sentences the verb of the matrix clause and the verb of the subclause are in our lexicon, and 6 of them involve negation. We included these cases into our test set and added 34 randomly chosen affirmative and negated sentences containing a single verb from the lexicon. For these sentences, we evaluated two different settings. In setting I, the treebank parses were used, in setting II the output of the ParZu dependency parser (Sennrich et al., 2013).

Table 8 shows the descriptive statistics for Test80 (column system showing the results for setting I)<sup>4</sup>. Two raters A and B independently annotated all test sentences according to simple guidelines that treat the prediction of the inferred effects and attitudes as a textual entailment task (Dagan et al., 2013).<sup>5</sup> After a reconciliation session only two cases had to be adjudicated by a third rater

<sup>4</sup>We cannot evaluate *MyProponent* and *MyOpponent* since these concepts depend on the individual preferences of the annotators.

<sup>5</sup>The annotators have to formulate factual entailment candidates that they then accept or reject. Given our running example sentence, they would typically create and check entailment sentences such as “Therefore, it is the case that the EU has a positive attitude towards Greece” for *pro*, or “Therefore, it is the case that the EU acts in a positive manner” for *benefactor*.



in order to establish a gold standard G. The pairwise agreement between A and B is 43% (Cohen’s  $\kappa = 0.19$ ), between A and G 69% ( $\kappa = 0.56$ ), B and G 61% ( $\kappa = 0.44$ ).  $\kappa$  between A and B is low, but this is mostly due to the difficulty of spotting candidate entities and relations in complex nested sentences, and not due to different annotation categories assigned to the same candidate. Humans are selective annotators and focus on the most striking attitudes more than on the more hidden ones. During reconciliation, missing annotations of one annotator could be easily spotted and adopted in view of the annotations of the other.

The overall performance of the system is 59.04% precision and 71.15% recall (setting I). If we replace perfect parse trees with parser output (setting II), precision is almost unaffected (58.84%), while recall drops to 50.64%.

We have identified some systematic errors of our system. Among others, it instantiates concepts from the effect layer (beneficiary etc.) too often, especially entities that are non-actors (e.g. “A criticizes the proposal” gives *victim(proposal)*). The gold standard only allows actors (person, company etc.) to occupy these roles. A better classification for actors would help in these cases.

A central claim of this paper is that factuality is important for sentiment inferences since it licenses or suppresses reasoning. Given our test set comprised of 80 sentences, 41 verb mentions were classified as non-factual and thus were blocked for certain inferences. If we switch off factuality detection (i.e., every verb is factual), a precision drop of 12.9% results (while recall increases only slightly by 1.2%).

## 9 Comparison with Deng & Wiebe

Recently, Deng and Wiebe (2014) and Deng and Wiebe (2015) have introduced an advanced conceptual framework for inferring sentiment implicatures. Their work is most similar to our approach. Various model versions exist, the latest one (Deng and Wiebe, 2015) also copes with event-level sentiment inference, which brings it even closer to our model. Probabilistic Soft Logic is used for the specification of the rules and for drawing inferences. The goal of the systems is to detect entity pairs that are in a PosPair or NegPair relation. This is similar to our pro/con relations.

First of all, factuality is not taken into account in their framework, while we have shown that it is

crucial for certain inference steps. Although their model is based on the idea of good/bad-for verbs, they do not envisage to propagate (as we do) such effects, i.e. determine whether these effects have occurred or not (clearly, factuality is crucial here). In contrast to our approach, their model is a probabilistic one. However, it is obviously not the layer of inference rules (the attitude projection in our terms) which establishes the source of uncertainty, it is the preprocessing where three existing sentiment systems and two SVM classifiers are used for polarity detection (i.e. identifying targets, polarity spans etc.). This obscures the fact that some inference rules might contribute to false predictions as well. For instance there is a rule (3.10 from Table 1, (Deng and Wiebe, 2015)) that more or less states that I am against any action of someone I do not like. Clearly, we hardly would be against a good deed of an opponent of us. We believe, though, that such over-generalized rules also exist in our model and that we should find a means to focus on that kind of failure (not so much on propagated errors from the preprocessing stages).

## 10 Conclusions

Our model strives to answer the following questions, given a text and the personal profile of a single user: who benefits (or suffers) from the situations described, what does the text (implicitly) tell us about the relationship of the actors involved, which topics does an actor like or dislike and – given all this – what does this imply for the user: who are proponents or opponents of his or hers.

The basis of our model is a language-specific verb polarity lexicon with polar effects on the bearers of what we call the *for-roles* and the *of-roles* of the verb. This and the predicate argument structures of a sentence lead to an A-Box representation of the sentence. OWL concepts and a set of SWRL rules then derive what the text implies about (the author’s view of) reality and what the reader might make of it.

## Acknowledgments

We would like to thank Noëmi Aepli and Don Tuggener for their support and the reviewers for their helpful comments. This work was conducted using the Protégé resource, which is supported by grant GM10331601 from the National Institute of General Medical Sciences of the United States National Institutes of Health.

## References

- Ido Dagan, Dan Roth, Mark Sammons, and Fabio Zanzotto. 2013. *Recognizing Textual Entailment: Models and Applications*, volume 6 of *Synthesis Lectures on Human Language Technologies*. Morgan & Claypool.
- Donald Davidson. 1966. The logical form of action sentences. In Nicholas Rescher and Alan Ross Anderson, editors, *The Logic of Decision and Action*, pages 81–95. University of Pittsburgh Press.
- Lingjia Deng and Janyce Wiebe. 2014. Sentiment propagation via implicature constraints. *Meeting of the European Chapter of the Association for Computational Linguistics (EACL-2014)*.
- Lingjia Deng and Janyce Wiebe. 2015. Joint prediction for entity/event-level sentiment analysis using probabilistic soft logic models. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 179–189.
- Birte Glimm, Ian Horrocks, Boris Motik, Giorgos Stoilos, and Zhe Wang. 2014. HermiT: An OWL 2 reasoner. *Journal of Automated Reasoning*, 53(3):245–269.
- Matthew Horridge, Nick Drummond, John Goodwin, Alan Rector, Robert Stevens, and Hai H Wang. 2006. The Manchester OWL syntax. In *OWL: Experiences and Directions (OWLED)*.
- Ian Horrocks and Peter F. Patel-Schneider. 2004. A proposal for an OWL rules language. In *Proc. of the Thirteenth International World Wide Web Conference (WWW 2004)*, pages 723–731. ACM.
- Ian Horrocks and Peter F. Patel-Schneider. 2011. KR and reasoning on the Semantic Web: OWL. In John Domingue, Dieter Fensel, and James A. Hendler, editors, *Handbook of Semantic Web Technologies*, chapter 9, pages 365–398. Springer.
- Lauri Karttunen. 2012. Simple and phrasal implicatives. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics - Volume 1: Proceedings of the Main Conference and the Shared Task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation, SemEval '12*, pages 124–131, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Manfred Klenner and Michael Amsler. 2016. Sentiframes: A resource for verb-centered german sentiment inference. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).
- Manfred Klenner, Michael Amsler, and Nora Hollenstein. 2014. Verb polarity frames: a new resource and its application in target-specific polarity classification. In *Proceedings of KONVENS 2014*, pages 106–115.
- Manfred Klenner. 2015. Verb-centered sentiment inference with description logics. In *6th Workshop on Computational Approaches to Subjectivity, Sentiment & Social Media Analysis*, pages 134–140, September.
- J. R. Martin and P. R. R. White. 2005. *Appraisal in English*. Palgrave, London.
- Alena Neviarouskaya, Helmut Prendinger, and Mitsuru Ishizuka. 2009. Semantically distinct verb classes involved in sentiment analysis. In Hans Weghorn and Pedro T. Isaías, editors, *IADIS AC (1)*, pages 27–35. IADIS Press.
- Kevin Reschke and Pranav Anand. 2011. Extracting contextual evaluativity. In *Proceedings of the Ninth International Conference on Computational Semantics*, pages 370–374.
- Roser Saurí and James Pustejovsky. 2009. FactBank: a corpus annotated with event factuality. *Language Resources and Evaluation*, 43(3):227–268.
- Rico Sennrich, Martin Volk, and Gerold Schneider. 2013. Exploiting synergies between open resources for german dependency parsing, pos-tagging, and morphological analysis. In *Recent Advances in Natural Language Processing (RANLP 2013)*, pages 601–609, September.
- Heike Telljohann, Erhard W. Hinrichs, Sandra Kübler, Heike Zinsmeister, and Kathrin Beck. 2009. Stylebook for the Tübingen treebank of written German (TüBa-D/Z). Technical report, Universität Tübingen, Seminar für Sprachwissenschaft.