Unifying Semantic Relations Across Syntactic Levels

Vivi Nastase and **Stan Szpakowicz**
School of Information Technology and Engineering
University of Ottawa
{vnastase, szpak} @site.uottawa.ca

Abstract

We construct a list of binary semantic relations that combines three lists recognized at different syntactic levels – multi-clause sentences, clauses, noun phrases. We identify relations that are the same at different levels and represent them in a level-independent way. We base our construction on the fact that many syntactic variants express the same idea, and a linguistic expression displaying a semantic relation at one level may be equivalently converted onto another level. Uniformity of representation allows a thorough text analysis, since the same relation can be recognized in many forms. A system trained to assign relations automatically will allow subsequent analysis of more complex semantic structures.

1 Introduction

We consider semantic relations that characterize the interaction between two occurrences[1], entities or attributes denoted by clauses and phrases. Our research project has previously identified and validated, through manual and semi-automatic experiments, lists of relations at the clause, intra-clause and noun-phrase level (Delisle 94), (Barker 98). These syntactic levels supply lexical and syntactic indicators we use to analyze semi-automatically and eventually automate the assignment of semantic relations. Semantic clues are derived from connectives and prepositions, and lexical knowledge bases (*WordNet* and *Roget's Thesaurus*). In a paragraph – one level above multi-clause sentences – syntactic clues are fewer and interactions between just two elements rare; we stop at the level of base noun phrase (modifier-noun pairs).

An idea can be expressed in many ways. The same semantic relation can appear at different syntactic levels; the corresponding linguistic expressions can be matched by compression or expansion. A combined list of relations has the advantages of generality and uniformity. Supervised machine learning will automate assigning semantic relations; it will help to have instances of a relation coming from more than one syntactic level.

[1]The term *occurrence* encompasses events, processes, actions, activities and accomplishments (Allen 84)

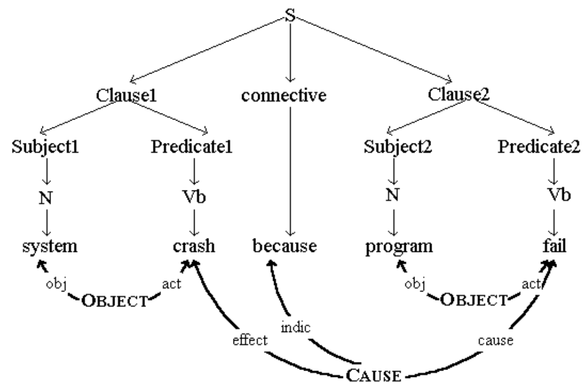

Figure 1: Semantic relations for the sentence "The system crashed because the program failed"

The semantic relations we work with, grouped into 6 classes, are presented in Table 1.

A semantic relation holds between certain types of entities or occurrences. Relations that intuitively convey the same meaning may be carried by different syntactic structures. A relation is the same if it links the same entities or occurrences regardless of the syntactic level. To prove this point, we need to analyse instances of the same (intuitively) semantic relation at different syntactic levels. In a collection of syntactically varied examples, we design for each relation a representation independent of the syntactic level, and test it by representing expressions from some corpus. Each relation on our unified list is represented by pointing at the constituents that interact in the way described by the relation (Fig.1). We do not discard any aspect of the original expression, just add a layer to the parse tree. The name of the structure characterizes mnemonically the interaction between constituents.

Phenomena that account for expression variation include compression and expansion. Several mechanisms play a role in compressing or expanding an expression to match its counterpart at another level.

Class	Relations	Class	Relation	Class	Relation
Causality	Cause	Quality	Container	Participant	Accompaniment
	Effect		Content (physical)		Agent
	Detraction		Topic		Beneficiary
	Enablement		Manner		Exclusion
	Entailment		Material		Experiencer - Stative
	Prevention		Measure		Experiencer - Property
	Purpose		Order		Experiencer - Possessor
Conjunctive	Conjunction		Equative		Experiencer - Possession
	Disjunction		Type		Instrument
Temporality	Co-occurrence	Space	Direction		Object
	Frequency		Location From		Object-property
	Precedence		Location To		Product
	Time At		Location Through		Recipient
	Time From		Location		Part
	Time Through		Located		Whole
	Time To		Orientation		

Table 1: The unified list of relations, grouped in 6 classes

2 Mechanisms for Expression Variation

2.1 Deletion

In passing from a detailed to a concise syntactic level, elements of the utterance may be dropped. (Levi 78) proposes nine *recoverable deletable predicates* (RDPs), five of which are verbs – *cause, have, make, use, be.* By deleting the verb the expression becomes a noun phrase, for example:

virus causes flu ⟶ *flu virus*

Levi has designed transformations that systematically change an expression by deleting RDPs. Depending on whether the *Agent* or the *Object* in the phrase *X causes Y* becomes the head noun, we obtain a noun phrase with the Cause :
$virus_{AGT}$ causes $flu_{OBJ} \leftrightarrow flu_{OBJ}\ virus_{AGT}$
or the Effect relation: $exam_{AGT}$ causes $anxiety_{OBJ} \leftrightarrow exam_{AGT}\ anxiety_{OBJ}$

The inverse transformation would have to recover the deleted information. This is much less obvious, but in the concluding remarks we suggest how to seek fillers for slots left empty by deletion.

2.2 Nominalization

Another compression mechanism is verb nominalization. The verb usually becomes the head of a noun phrase. (Quirk *et al.* 85) mention that one can map the verb's arguments onto the modifiers of the deverbal noun in the corresponding noun phrase. The concepts are the same (the surface realization has changed), so the noun-modifier relations can also be mapped onto the case roles. This is also the assumption in (Hull & Gomez 96), where one of the tasks is assigning thematic roles to the modifiers of a deverbal head noun, according to the underlying verb.

"Inverse nominalization" would recover the verb from its nominal form. There is no direct algorithm. We work with a list of ⟨nominal form,verb⟩ pairs, semi-automatically extracted from texts. It would help to use *WordNet* or rather *Roget's Thesaurus* for links between noun and verb senses. In (Hull & Gomez 96) deverbal nouns are those with *action* as a hypernym.

2.3 Adjectivalization

In the example:

the nation has a large debt

we adjectivalize *nation* to obtain *national debt.*

A past participle can become the modifier in a noun phrase, as in *vanished treasure*

The inverse operation restores the verb form of the word. For example:

vanished treasure → *the treasure vanished*

2.4 Synecdoche

Will be illustrated in the example on **Causality** .

2.5 Equivalence

Will be illustrated in the example on **Temporality** .

2.6 Relations Across Levels

(Givon 75), in a deeper analysis of causality, transforms a multi-clause sentence into a clause, assuming *weak relatedness* between the results of the transformation stages. In his view, a causal relation that held between propositions now holds between specific parts of the propositions:

P_c [cause] P_e

Agt_{Pc} [cause] P_e by P_c

Agt_{Pc} [cause - Vb_{Pe}] $Patient_{Pe}$ by Vb_{Pc}

In the following example:
George shot the gun at the elephant, and as a result, the elephant died.
Agt_{Pc} is George – the real cause of the elephant's death, as the agent who initiated the action.

This view suggests that it is rather the *Agent* in the *Cause* proposition that is the real cause of the action in the *Effect* proposition P_e, which has an effect on the *Patient* in P_e (Givon's PATIENT corresponds to our OBJECT).

While we think that in this particular example causality relations hold between two occurrences, we agree that sometimes clause-level relations are better explained by considering clause components other than the head verb. We will see that this seems to be true especially for the semantic relations in the **Causality** group, but the clause-level relations grouped under **Temporality** are clearly defined by the complete clauses.

3 Two Examples

Clause-level relations hold between two clauses. The head of a clause is the verb, so in principle the relation holds between the occurrences described by verbs. This however depends on the relation. **Temporal** relations[2] respect this condition. They hold between the underlying actions or events, expressed by the head verbs of the two clauses. For **Causality** , this is not necessarily true. For example:

The file printed because the program issued a command.

The clauses are in the CAUSE relation, although *printed* was not caused by *issued*. *issued* merely causes *command* to appear, which will cause *printed* to occur (by an unspecified action).

CAUSE and EFFECT are very similar. If the emphasis is on the *Cause* part of the relation, the relation is CAUSE , otherwise it is EFFECT . For example flu_{EFF} $virus_{CS}$ (CAUSE) versus $viral_{CS}$ flu_{EFF} (EFFECT).

Our representations result from a manual analysis of examples acquired manually from (Barker 98), (Levi 78), and automatically from (Larrick 61) and the Brown corpus. We designed additional examples to display better certain phenomena. We manually annotated with semantic relations all the examples from our list, except those in (Barker 98).

3.1 Causality - The Effect relation

EFFECT links two occurrences, a *Cause* and an *Effect* . The *Cause* is usually an event whose unfolding causes another event or a state to occur. As Givon argues, a relation is sometimes better described by specific participants than by the occurrence itself. For example:

The student was anxious because he was writing an exam.

The *writing* causes the student to *be anxious.* We associate the *exam* with some actions, but if we do not mention the action, the exam's effect on the student will be the same:

The student was anxious because of the exam.

The *exam* is now the cause of anxiety. The underlying action can even occur in the future – the important part is the *exam*. Now, if exams usually make people anxious, we can generalize:

exam anxiety

The *Effect* now is a state of anxiety, caused by an unspecified action involving exams.

The examples illustrate the EFFECT relation with a varying amount of detail. We want to represent all instances of EFFECT in the same way, regardless of their syntactic level. A slot may remain empty if some information is missing. The examples suggest that sometimes the actual *Cause* occurrence is only *implied* by one of its parts – the *Object* in *exam anxiety*, the *Agent* in *viral flu*. We consider this a phenomenon similar to synecdoche: we look at the *Agent* , *Object* and other arguments of the main verb as part of the occurrence. The whole is replaced by a part.

To represent EFFECT , the structure should point to the *Cause* occurrence, the *Effect* occurrence, and the INDICATOR . A pointer to an occurrence gives access to its every attribute and argument. Our examples show, however, that the occurrence is not always presented using a full clause. Sometimes a part stands for the whole. We will incorporate this in the structure we design. We have also found empirically that the *Effect* occurrence is not replaced by its part. The *Effect* element will therefore be a pointer to the head of the occurrence – a verb or a deverbal noun.

We propose a slot filler structure presented in Table 2 for the EFFECT relation illustrated in our examples. The lexeme fillers should be regarded as pointers to those lexemes in the syntactic representation of the utterance.

We can extract from these representations (and from other examples) the common structure of the EFFECT relation, presented in Table 2.

We note that some fillers are empty. This is not a downside. Our representation encourages

[2] We write the names of relation classes bold (**Causality**), relations as small caps (CAUSE), and relation argument names italicized (*Cause* for "virus" in "flu virus")

Example	CAUSE	EFFECT	INDICATOR
The student was anxious because he was writing an exam.	OCCUR. [VB./STATE write; OCCUR. PART [TYPE OBJECT; FILLER exam]]	OCCUR. [VB./STATE be anxious]	*because*
The student was anxious because of the exam.	OCCUR. [VB./STATE -; OCCUR. PART [TYPE CAUSE; FILLER exam]]	OCCUR. [VB./STATE be anxious]	*because*
exam anxiety	OCCUR. [VB./STATE -; OCCUR. PART [TYPE -; FILLER exam]]	OCCUR. [VB./STATE anxiety]	-
	OCCUR. [VB./STATE `pointer to the cause occurrence`; OCCUR. PART [TYPE `occurrence element`; FILLER `pointer to the occurrence element`]]	OCCUR. [VB./STATE `pointer to the effect occurrence`]	`indicator`

Table 2: Proposed representation for the EFFECT relation

interesting experiments to find possible fillers for the empty slots. Indicators are an obvious example. For the VERB/STATE filler, a class of verbs (or deverbal nouns) can be associated with that particular occurrence part (*Object* / *Agent*), such that this action causes the mentioned effect.

3.2 Temporality

Temporal relations hold between two time intervals. According to the actual span of these time intervals and their relative position on the time axis, we obtain different relations (Allen 84). An interval can be expressed not only by an explicit time expression, but also by an occurrence unfolding in time. We can thus find the same **Temporal** relation at different syntactic levels.

Table 3 shows three examples of the TIME THROUGH relation and the common structure.

In (Nastase 01) there is a more detailed presentation of each of the relations in our unified list, with the proposed representation structure.

4 Validating the Representations

The representation structures we proposed were validated on our collection of examples. The algorithm that follows shows a semi-automatic way of validating the patterns and the unified list of relations. It hurts little if some noun phrases are missed.

Example	OCCUR.	INTERVAL	INDICATOR
The band practices while others have lunch.	[VB./STATE practice]	[TYPE OCCUR.; FILLER [VB./STATE have lunch]]	while
The band practices during lunch hour.	[VB./STATE practice]	[TYPE DEFINITE INTERVAL; FILLER lunch hour]	during
lunch-hour practice	[VB./STATE practise]	[TYPE DEFINITE INTERVAL; FILLER lunch hour]	-
	[VB./STATE `occurrence 1`]	[TYPE `type of interval`; FILLER `interval`]	`indicator`

Table 3: Representation for TIME THROUGH

- Tag text $\mathcal{T}$ using a part-of-speech tagger. (We used Brill's public-domain tagger)
  ... the/DT plans/NNS for/IN their/PRP$ brick/NN house/NN by/IN the/DT river/NN ...

- Let $\mathcal{P}$ be a set of modifier-noun pairs from $\mathcal{T}$ obtained by sliding a four-word window w_1, w_2, w_3, w_4. w_2, w_3 is a modifier-noun pair if w_3 is noun, w_2 is noun, adjective, or adverb, and w_1 and w_4 are not noun, adjective or adverb.
  ... plans/NNS for/IN [**their/PRP$ brick/NN**

house/NN by/IN] *the/DT river/NN ...*
⟶ *brick/NN house/NN* - is a modifier-noun pair (base noun-phrase)

- For each modifier-noun pair *np* in $\mathcal{P}$:
  - find paraphrases in $\mathcal{T}$. A paraphrase of *np* is an expression other than a modifier-noun pair, which contains two words derived from the two words in *np*.
    ... they build **houses** *with* **bricks** *...*
  - assign semantic relations to pairs of entities in the paraphrase
  - map the paraphrase onto the corresponding structure

The available corpora, including Brown, usually contain numerous short texts. It is not likely to find many paraphrases, as a study we performed on the Brown has shown. It could be more likely to find paraphrases by using the Internet as a corpus. For a given modifier-noun pair we can determine words derived from the modifier and the noun, and give these words to a search engine. The use of the Internet as a resource was introduced by (Mihalcea & Moldovan 99), to acquire statistical measurements of word co-occurrences.

5 Conclusions and Future Work

We will test the validity of the unified list of relations and the proposed representation structures in a semi-automatic system that uses machine learning to build a compact representation of a document, annotated with syntactic and semantic information. The structures will play a role in assigning semantic relations to pairs of entities. Independence of the syntactic level increases the chance of finding a previously annotated example to match the one under analysis.

A slot in a pattern may be empty. This means that the listener can infer the filler from what has been already said. We can analyze possible transformations, recoverable deletable predicates (in the spirit of (Levi 78)) or classes of entities (by generalizing in an ontology) – which should fill empty slots.

Our representations could be used to collect patterns and help prove or disprove our hypotheses about the existence of systematic transitions between syntactic levels. We can postulate operations that, when applied to an expression, produce a semantically related expression at a different syntactic level, and then test them using our representation by comparing patterns. Semantic relations may subcategorize for classes of entities just as verbs subcategorize for arguments. The analysis of patterns extracted from text could bring evidence to support or reject this hypothesis. We do not argue that such operations can or should be applied to all expressions.

We also want to use this representation to analyze the change of semantic relations when the utterances are changed by deletion, as discussed in Section 2. If a verb is deleted, we want to see how the case relations change, and what they correspond to in the newly formed noun phrase.

Text analysis aims to represent the knowledge contained in the text so that it can be accessed and used for reasoning, learning and other purposes. A unified list of relations will help build a more concise representation of the text, because concepts and relations between them will be identified regardless of the surface form in which they appear. Concepts will not have to be duplicated, and only new links between them will be added. Accessing a node in such a representation will give access to a variety of syntactic and semantic information in a format that supports processing.

6 Acknowledgements

Partial funding for this work comes from NSERC.

References

(Allen 84) J. F. Allen. Towards a General Theory of Action and Time. *Artificial Intelligence*, 23:123–154, 1984.

(Barker 98) K. Barker. *Semi-Automatic Recognition of Semantic Relationships in English Technical Texts.* Unpublished PhD thesis, Department of Computer Science, University of Ottawa, 1998.

(Delisle 94) S. Delisle. *Text Processing Without a-priori Domain Knowledge: Semi-Automatic Linguistic Analysis for Incremental Knowledge Acquisition.* Unpublished PhD thesis, Department of Computer Science, University of Ottawa, 1994.

(Givon 75) T. Givon. Cause and Control. *Syntax and Semantics*, 4:59–89, 1975.

(Hull & Gomez 96) R.D. Hull and F. Gomez. Semantic Interpretation of Nominalizations. In *Proc. Thirteenth National Conference on AI*, pages 1062–1068, Portland, Oregon, USA, 1996.

(Larrick 61) N. Larrick. *Junior Science Book of Rain, Hail, Sleet and Snow.* Garrard Publishing Company, Champaign, IL, 1961.

(Levi 78) J. N. Levi. *The Syntax and Semantics of Complex Nominals.* Academic Press, NY, 1978.

(Mihalcea & Moldovan 99) R. Mihalcea and D. I. Moldovan. A Method for Word Sense Disambiguation of Unrestricted Text. In *Proc. 37th annual meeting of* *the ACL*, pages 152–158, College Park, Maryland, USA, 1999.

(Nastase 01) V. Nastase. Unifying Semantic Relations Accros Syntactic Levels http://www.site.uottawa.ca/~vnastase/papers/TR-2001-02.pdf. Technical report, University of Ottawa, SITE, 2001.

(Quirk *et al.* 85) R. Quirk, S. Greenbaum, G. Leech, and J. Svartvik. *A Comprehensive Grammar of the English Language*. Longman, London and New York, 1985.