



UvA-DARE (Digital Academic Repository)

Making sense of legal texts

de Maat, E.

Publication date
2012

[Link to publication](#)

Citation for published version (APA):

de Maat, E. (2012). *Making sense of legal texts*. [Thesis, fully internal, Universiteit van Amsterdam].

General rights

It is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), other than for strictly personal, individual use, unless the work is under an open content license (like Creative Commons).

Disclaimer/Complaints regulations

If you believe that digital publication of certain material infringes any of your rights or (privacy) interests, please let the Library know, stating your reasons. In case of a legitimate complaint, the Library will make the material inaccessible and/or remove it from the website. Please Ask the Library: <https://uba.uva.nl/en/contact>, or a letter to: Library of the University of Amsterdam, Secretariat, Singel 425, 1012 WP Amsterdam, The Netherlands. You will be contacted as soon as possible.

5 Creating Model Fragments

After each sentence has been classified, an actual model for each sentence can be constructed.

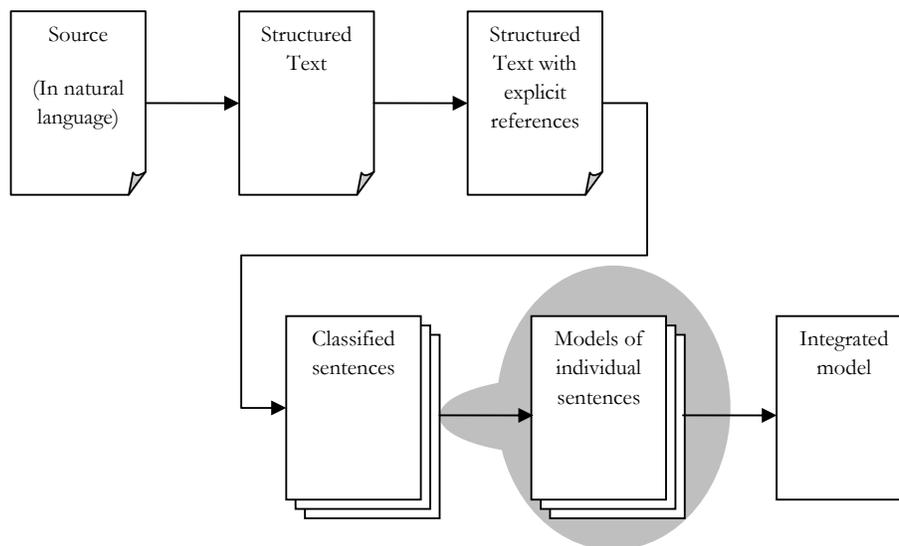


Figure 10: Modelling sentences step

The input for this step consists of classified individual sentences, with references marked-up. We also know the pattern that was used to classify each sentence. For several types of sentences, these patterns, together with some added features, are sufficient to extract all information needed to create a model of the sentence. This is usually the case with sentences that are about the law itself, instead of the subject matter of the law. These sentences are discussed in section 2. Other sentences, such as obligations, do focus on the subject matter, and can vary wildly. Simple patterns will not suffice to deal with these sentences, and to extract information from these types of sentences, we use a Dutch grammar parser. These sentences are discussed in section 3. Several of the ideas in this chapter have been published before (de Maat & Winkels, 2010a, 2011a, 2011b).

There have been earlier attempts to convert parse trees to models. Bos et al. (2004) have converted the output of a parser for the English language to first order logic using lambda calculus. An example of such a conversion, given by Bos et al., is:

The school-board hearing at which she was dismissed was crowded with students and teachers

This results in the following first-order logic representation:

```


$$\exists a((\text{school-board}(a) \ \& \ \text{hearing}(a)) \ \& \ \exists b(\text{female}(b) \ \& \ \exists c(\text{dismiss}(c) \ \& \ \text{patient}(c,b) \ \& \ \text{at}(a,c) \ \& \ \exists d(\text{crowd}(d) \ \& \ \text{patient}(d,a) \ \& \ ((\exists e(\text{student}(e) \ \& \ \text{with}(d,e)) \ \& \ \exists f(\text{teacher}(f) \ \& \ \text{with}(d,f))) \ \& \ \text{event}(d))))))))$$


```

A similar approach has been followed by McCarthy (2007), who uses unification to transform the parse tree to a quasi-logical form. He has applied this technique on judicial opinions from

- travelsAtLeastOnceAWeekBetweenResidenceAndPlaceOrPlacesOfWork
- travellingBackAndForthBetweenResidenceAndPlaceOfWorkIsDoneWithin24Hours

In IBM's translation of The Freedom of Information and Protection of Privacy Act (Powers, Adler & Wishart, 2004) we find similar coarse-grained attributes:

- proper-exceptions-to-notice-obligation
- disclosure-by-law-enforcement-institution
- authorized-by-a-constituent-next-of-kin-or-legal-representative
- for-the-purpose-for-which-it-was-obtained-or-compiled-or-for-a-consistent-purpose

These attributes all correspond with a single question that can be communicated to the user. Ideally, we want to generate models that are more in line with these attributes and conditions. Sarwa Bajwa, Samad and Mumtaz (2009) use an approach that creates UML models by (broadly speaking) mapping nouns to classes, verbs to methods and adjectives to attributes. Thus, the sentence:

Customer has purchased a red ball.

Results in the following model:

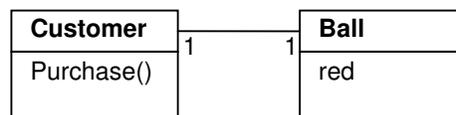


Figure 12: Samand and Mumtaz' UML model for *Customer has purchased a red ball*.

A similar approach has been attempted as part of the POWER project, where partial UML models were generated by creating classes based upon nouns found in the text and adding adjectives and adjective phrases as attributes. Several “fixed juridical constructs” (many of which correspond with the patterns used by us to classify sentences) can also be recognised and are translated as a specific rule attached to the class (van Gog & van Engers, 2001). For example, the sentence:

a house is considered an owned house if the house is owned by the tax payer

Results in the model:

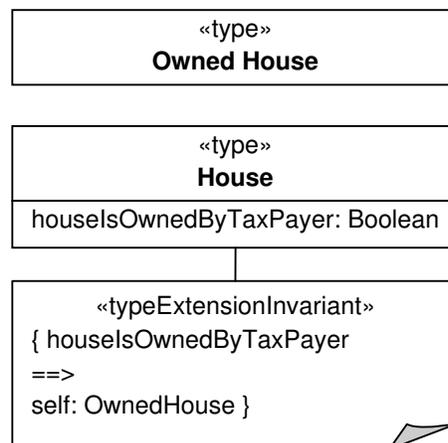


Figure 13: Van Gog and van Engers' UML model for a house is considered an owned house if the house is owned by the tax payer

Other methods do not generate UML models, but mark-up relevant sections of the text. Kiyavitskaya et al. (2008) use similar patterns as presented in section 4.1 to identify rights and obligations in case law and to identify certain basic entities: *actor*, *policy*, *event*, *date* and *information*. These were then marked in the text, yielding results such as:

```
<Right>A <Actor>covered entity</Actor> may deny an <Actor>individual</Actor>'s
request for amendment,</Right> if it determines that the <Information>protected health
information</Information> or record that is the subject of the request:
<Index>(i)</Index> Was not created by the <Actor>covered entity</Actor>,
<Exception>unless the <Actor>individual</Actor> provides a reasonable basis to believe
that the originator of <Information>protected health information</Information> is no
longer available to <Policy>act</Policy> on the requested amendment </Exception> ...
```

For statute law, a similar approach has been followed by Biagioli, Francesconi, Passerini, Montemagni and Soria (2005), who attempt to fill in frames for provisions in Italian laws. For example, for an obligation, this approach attempts to fill the slots addressee, action and third-party. It does so by assigning specific elements of the sentence to specific slots. As an example of this, they say: “the addressee of an obligation typically corresponds to the syntactic subject of the sentence, while the action (s)he is obliged to carry out is usually expressed as an infinitival clause”. This leads to mark-up such as:

```
[[[Il comitato misto]subj]addressee] `e tenuto [[a raccomandare modifiche degli allegati
secondo le modalit`a previste dal presente accordo]i_clause]action]

[[[The Joint Committee]subj]addressee] shall [[be responsible for recommending
amendments to the Annex as foreseen in this Agreement]i_clause]action].
```

Brighi, Lesmo, Mazzei, Palmirani and Radicioni (2008) and Palmirani and Brighi (2010) likewise try to fill semantic frames by mapping parts of a parse tree to roles in the frame. They confine themselves to amending clauses in Italian law. A similar system was made by Spinosa et al. (2009), though they used a chunker, not a parser, as a pre-processor for their system¹¹⁷.

For many sentences, though, the use of a parser to recognise the different elements in a sentence does not seem to be necessary. In particular, sentences that deal with the law itself (rather than with the application domain of the law) follow such strict patterns, that the information can be extracted based upon these patterns. This is demonstrated by Ogawa, Inagaki and Toyama (2008), who obtain a formal representation of amending clauses by means of regular expressions.

In this chapter, we will attempt to create semantic frames for Dutch law, similar to those introduced by van Kralingen (1995) en used by Biagioli et al. (2005) and Brighi et al. (2008). In order to fill these frames, we will use textual patterns for those sentences that deal with the law (discussed in section 5.1, followed by an experiment in section 5.2). For those sentences that deal with the application domain of the law, we will use a parser to fill the slots (discussed

¹¹⁷ A chunker splits a sentence into groups of related words, such as a noun phrase. It provides less information than a full parser.

in section 5.3, followed by a discussion of our experiences with this method in section 5.4). The next step in the process, the integration of the models, is discussed in section 5.5.

5.1 Sentences Dealing With the Law

There are several types of sentences that are about the law itself. These are sentences that change the text of a law or set the enactment date of a law. Compared to sentences which deal with the subject of the law (which could be anything: oil tankers, accountancy, book prices, etc.) they have very limited variation. In general, they follow clear patterns, from which it is easy to extract the information we need.

5.1.1 Scope Declaration

A scope declaration always contains one relevant piece of information: the scope. This scope is identified by a reference to the relevant text. If we assume that the reference parser has successfully identified all references during pre-processing, then the scope is easily found: it is the reference at the start of the sentence, or more easily: the one reference that appears in the sentence.

Act of May 20th, 2010 (Stb. 2010/200), article I, introduction¹¹⁸

The Penal Code is modified as follows:

Scope	
Scope	Penal Code

If a scope is set for an article or subpart, then all changes made within that article or subpart refer to that scope.

5.1.2 Repeal

A repeal removes an entire law, or part of that law. Other than the knowledge that we are dealing with a repeal, only one bit of information needs to be extracted: the reference to the text that is to be repealed. This reference is directly at the beginning of the main sentence (though it is also the only reference).

Act of June 12th, 2008 (Stb. 2008/229), article I, sub B¹¹⁹

Article 4 is repealed.

Repeal	
Target	Article 4

Alternatively, only part of the text is repealed, i.e. a number of words are deleted. In that case, we need to extract the location where the text to be modified is located, and the text that should be removed. The location will be a reference at the start of the sentence, following the word *In* (though again, it will be the only reference in the sentence). The text to be removed will be marked with double angle quotes.

¹¹⁸ **Wet van 20 mei 2010 (Stb. 2010/200), artikel I, aanhef**

Het Wetboek van Strafrecht wordt als volgt gewijzigd:

¹¹⁹ **Wet van 12 juni 2008 (Stb. 2008/229), artikel I, lid B**

Artikel 4 vervalt.

Act of May 20th, 2010 (Stb. 2010/205), article III¹²⁰

In article 438, sub 1, 2nd item, of the Penal Code, «profession or appointment, » is repealed.

Repeal	
Target	Penal Code, article 438, sub 1, 2 nd item
Text	profession or appointment,

5.1.3 Insertion

An insertion inserts some new text into the document. There are several different scenarios for insertion. The easiest situation exists when an entire new (numbered) element is inserted. In this case, sentence will be something like this:

Act of May 13th, 2004 (Stb. 2004/220), article I, sub A¹²¹

After article 7:1 a new article is inserted in section 7.1, reading: ...

We need to model two pieces of information. Firstly: the text to be inserted. This can be found after the colon. Then, we need to know where to insert it. The (part of the) document is denoted by a reference that is preceded by *in* (in this case: *in section 7.1*). If the insertion has been preceded by a scope declaration, then this reference may be incomplete, and needs to be combined with the scope declaration to get a complete reference. The location is given by another reference, preceded by either *before* or *after*¹²².

Insertion	
Target	Section 7.1
Location	Article 7:1
Position	after
Text	...

If the target is entirely defined by the scope definition, it will be missing from the sentence. Similarly, if the location is the end of the document, no reference for the position will be given. Instead, the text is said to be *appended* instead of *inserted*. For example, the following sentence is missing both references:

Act of May 20th, 2010 (Stb. 2010/200), article I, sub A, sub 2¹²³

A paragraph is appended, reading: ...

Insertion	
Position	end
Text	...

If a modification is made within a paragraph, it is not possible to point to the position using a reference (as sentences and words are not numbered). In these cases, instead of a reference, quoted text is used to describe the location:

¹²⁰ **Wet van 20 mei 2010 (Stb. 2010/200), artikel III**

In artikel 438, eerste lid, onder ten 2e, van het Wetboek van Strafrecht vervalt «beroep of betrekking.».

¹²¹ **Wet van 13 mei 2004 (Stb. 2004/220), artikel I, lid A**

Na artikel 7:1 wordt in afdeling 7.1 een artikel ingevoegd, luidende: ...

¹²² More complex positioning sometimes occurs, but will not be discussed here.

¹²³ **Wet van 20 mei 2010 (Stb. 2010/200), artikel I, lid A, lid 2**

Er wordt een lid toegevoegd, luidende: ...

Act of May 20th, 2010 (Stb. 2010/200), article VII¹²⁴

In article 30, sub 6, of the Motorised Vehicles Liability Insurance Act, a sentence is inserted after the sentence «...», reading: ...

The quoted text is marked using double angle quotation marks.

Insertion	
Target	Motorised Vehicles Liability Insurance Act, article 30, sub 6
Location	...
Position	After
Text	...

5.1.4 Replacement

There are two main patterns of replacement, which yield a slightly different frame. The first pattern deals with the replacement of text within a given structure part. The text to be replaced is always marked with double angle quotation marks «». The replacing text is always at the end, preceded by a colon. The location of the replacement is at the beginning of the sentence, a reference following the word *In*. As with the other changes, this should be the only reference in the main sentence (disregarding any references that appear inside the quoted text).

Act of May 20th, 2010 (Stb. 2010/200), article I, sub B¹²⁵

In article 43a, «a conviction» is replaced by: an earlier conviction.

Replacement	
Target	article 43a
Old Text	a conviction
New Text	an earlier conviction

The second type of replacement replaces an entire structure element, and does therefore not include a text to replace. The replacing text is again found after the colon, and the location is given as a reference after *In* at the beginning of the sentence. The location may be missing, which is usually the case if it has been set earlier with a scope declaration.

Act of May 20th, 2010 (Stb. 2010/205), article IV, sub I¹²⁶

Article 31, sub 2, will read:

2. ...

Replacement	
Target	Article 31, sub 2
New Text	2. ...

¹²⁴ **Wet van 20 mei 2010 (Stb. 2010/200), article VII**

In artikel 30, zesde lid, van de Wet aansprakelijkheidsverzekering motorrijtuigen wordt na de zin «...» een zin ingevoegd, luidende: ...

¹²⁵ **Wet van 20 mei 2010 (Stb. 2010/200), artikel I, lid B**

In artikel 43a wordt «een veroordeling» vervangen door: een vroegere veroordeling.

¹²⁶ **Wet van 20 mei 2010 (Stb. 2010/205), artikel IV, lid I**

Artikel 31, tweede lid, komt te luiden:

2. ...

5.1.5 Renumbering

Renumbering changes the index of parts of a text. Though it is very uncommon for articles and document parts above the level of articles to be renumbered, paragraphs and list items are frequently renumbered. The sentence will list the part(s) to renumber and the new numbering. A reference to the part(s) to renumber can be found at the start of the sentence, while the new index can be found after *are renumbered to*.

Modern Migration Policy Act, article VIII, sub B¹²⁷

The articles 2a to 2u are renumbered to the articles 2i to 2cc.

Renumbering	
Target	articles 2a to 2u
New numbering	articles 2i to 2cc

A second common pattern for renumbering is adding a number to a previously unnumbered paragraph, such as:

Care Contract System Revision Act, article II, sub Q, sub 1¹²⁸

Before the text of article 64, the index "-1." is placed.

This operation is different from the “real” renumbering, as it changes the structure of the article by introducing a subparagraph. Hence, it cannot use the frame shown above, though the fields are very similar. This operation has a target, which may be present in the sentence as a reference following *the text of*. In the case of our example, this target is *article 64*¹²⁹. It is more common, though, for this kind of sentence not to name the target, which will have been declared before through a scope provision, and will have to be retrieved from the scope provision when integrating the models¹³⁰. The other field of the frame is the new index. So, for the example above, the complete frame would be:

Numbering	
Target	article 64
New index	-1.

5.1.6 Enactment Date

The basic sentence that sets an enactment date will include two pieces of information: the document (or parts of document) to be enacted, and the date on which it is to be enacted.

Fuel Taxes Environment Tariff Act 1991, article IV, sub 1, first sentence¹³¹

This law is enacted starting on January 1st, 1991.

¹²⁷ **Wet modern migratiebeleid, article VIII, lid B**

De artikelen 2a tot en met 2u worden vernummert tot de artikelen 2i tot en met 2cc.

¹²⁸ **Wet herziening overeenkomstenstelsel zorg, artikel II, lid Q, lid 1**

Voor de tekst van artikel 64 wordt de aanduiding "-1." geplaatst.

¹²⁹ Strictly speaking, the target of the operation is *the text of article 64*, but as this operation is always aimed at the text, simply identifying it as *article 64* will survive.

¹³⁰ Again, strictly speaking, such sentences do name the target as *the text*, which is an incomplete reference that can be completed using the scope provision (like other incomplete references that are governed by a scope provision).

¹³¹ **Tarievenwet brandstofheffingen milieu 1991, artikel IV, eerste lid, eerste volzin**

Deze wet treedt in werking met ingang van 1 januari 1991.

Enactment Date	
Target	this law
Date	January 1st, 1991

It is rather uncommon for a law to specify the exact date on which it will be enacted. More commonly, it will defer the decision to a royal decree:

Exception Situations Coordinating Act, article 11¹³²

This law is enacted on a date to be set by Royal Decree.

Delegated Enactment Date	
Target	this law
Delegation target	Royal Decree

Or it will relate it to the publication date:

Act of July 7th, 2010 (Stb. 2010/305), article 9¹³³

This law is enacted starting on the day after the date of publication of the Bulletin of Acts and Decrees in which she is included.

Enactment Date	
Target	this law
Date	the day after the date of publication of the Bulletin of Acts and Decrees in which she is included.

Both constructions follow fixed text patterns, which can be detected instead of a date. In addition to these straightforward enactment date setting sentences, more complicated constructions exist which include exceptions for which the date is set separately.

5.1.7 Short Title

A sentence setting a short title contains two pieces of information: the law for which the short title is set, and the actual short title.

Nobility Act, article 9¹³⁴

This law may be cited as Nobility Act.

Or:

Trade Name Act, article 10¹³⁵

This law may be cited under the title "Trade Name Act".

¹³² **Coördinatiewet uitzonderingstoestanden, artikel 11**

Deze wet treedt in werking op een bij koninklijk besluit te bepalen tijdstip.

¹³³ **Wet van 7 juli 2010, artikel 9**

Deze wet treedt in werking met ingang van de dag na de datum van uitgifte van het Staatsblad waarin zij wordt geplaatst.

¹³⁴ **Wet op de adeldom, artikel 9**

Deze wet kan worden aangehaald als Wet op de adeldom.

¹³⁵ **Handelsnaamwet, artikel 10**

Deze wet kan worden aangehaald onder de titel "Handelsnaamwet".

The law concerned is indicated with a reference at the start of the sentence. Usually, this is *this law*, but sometimes, the short title of some other law is changed. The new short title follows the text *may be cited as*, *may be cited under the title* or *may be cited under the name*. It may be enclosed in double or single quotes, and is sometimes preceded by a colon.

Short title	
Target	This law
Short title	Trade Name Act

5.1.8 Application Provision

An application provision refers to a text that does (or does not) apply in a given context (usually the article in which the application provision). For the most basic application provisions, there is only one piece of information to be extracted: a reference to the text that does or does not apply. This reference can be found at the beginning of the sentence:

Sickness Benefits Act, article 11a, sub 5¹³⁶
Article 11, sub 3, applies accordingly.

Application	
Target	Article 11, sub 3

Many application provisions do come with additional restrictions, which often relate to the subject matter of the law.

Partnership Taxes Act 1969, article 12b, sub 4¹³⁷
This article does not apply to brands and logos produced by the taxable person and any similar assets.

If this is the case, then the core of the sentence is still modelled in the same manner, but the additional restrictions need to be processed in the manner presented in section 3.

5.1.9 Delegation

A delegation contains two pieces of information: to whom or what the setting of rules is delegated, and what the scope of such rules may be. For example:

Notary Act, article 56, sub 5¹³⁸
By order-in-council, rules are set regarding the manner in which the size of the personal capital as meant in sub 4 is determined.

Here, the setting of new rules is delegated to an order-in-council, and their subject is *regarding the manner in which the size of the personal capital as meant in sub 4 is determined*. We can assemble the following frame:

¹³⁶ **Ziektewet, artikel 11a, vijfde lid**

Artikel 11, derde lid, is van overeenkomstige toepassing.

¹³⁷ **Wet op de vennootschapsbelasting 1969, artikel 12b, vierde lid**

Dit artikel is niet van toepassing op door belastingplichtige voortgebrachte merken, logo's, en daarmee vergelijkbare vermogensbestanddelen.

¹³⁸ **Notariswet, artikel 56, vijfde lid**

Bij algemene maatregel van bestuur worden regels gesteld met betrekking tot de wijze waarop de hoogte van het in het vierde lid bedoelde eigen vermogen wordt bepaald.

Delegation (Obligation) ¹³⁹	
Target	order-in-council
Subject	the manner in which the size of the personal capital as meant in sub 4 is determined

As the sentence does not contain an agent, the frame does not contain one either. As a result of integration with other rules (in this case, generic rules which restrict who is allowed to issue an order-in-council), an agent can be added.

The subject usually follows words like *regarding* or *on*. In the case of a sentence starting with *by order-in-council*, the target is always an order in council. In an active sentences, with a pattern like *may set rules* or *sets rules*, the subject of the sentence (appearing just before the pattern) is the target:

Sworn-in Interpreters and Translators Act, article 15, sub 2¹⁴⁰
Our Minister sets rules with regards to the certificate.

Delegation (Obligation)	
Target	Our Minister
Subject	the certificate

5.1.10 Publication Provision

A publication provision mentions a decision or decree that has to be published, who is responsible for publishing it, and where it should be published. For example:

Consumer Protection Retainment Act Article 5.1, sub 5¹⁴¹
Our Minister announces the cooperation protocols in the State Gazette.

In this pattern, the person making the announcement is listed before the pattern (*announces*), the subject is listed after the announcement, and the venue is listed following *in*.

Publication	
Agent	Our Minister
Venue	the Staatscourant
Subject	the cooperation protocols

An alternate pattern uses a passive sentence, which does not list the agent:

Announcement Act, article 4, sub 2, sentence 2¹⁴²
Such an announcement is announced in the State Gazette.

¹³⁹ The header *Delegation (Obligation)* refers to the fact that the rules must be set, rather than a situation in which rules *may* be set (which we would denote as a *Delegation(Permission)*).

¹⁴⁰ **Wet beëdigde tolken en vertalers, artikel 15, tweede lid**

Onze Minister stelt regels ten aanzien van het legitimatiebewijs.

¹⁴¹ **Wet handhaving consumentenbescherming, artikel 5.1, vijfde lid**

Onze Minister doet mededeling van de samenwerkingsprotocollen in de Staatscourant.

¹⁴² **Bekendmakingswet, artikel 4, tweede lid, tweede volzin**

Van zodanige bekendmaking wordt mededeling gedaan in de Staatscourant.

Here, the subject is listed before the pattern (*is announced*), and the resulting frame is:

Publication	
Venue	the State Gazette
Subject	such an announcement

5.2 Experiment

In order to determine the feasibility of the modelling rules as prescribed in the previous sections, we have conducted a study of the sentences of the corpus that was also used in the experiment described in section 4.5. For each sentence belonging to one of the categories that we wish to model, it was determined by hand whether or not the rules presented in the previous sections could actually be applied in order to decompose the sentence into the different elements needed for the model.

Once again, this is about patterns. However, these patterns are not the ones used to classify the sentences (as presented in chapter 4). Here, we assume that the sentence has been classified correctly (even though that will not always be the case), and apply different patterns; hence the results differ from those in chapter 4. The outcome of this experiment is shown in table 17.

	Sentences	Lists	No problems	Missing references	Additional parsing required	Both missing references and additional parsing required	Patterns do not apply
Delegation	19	4	18				5
Publication provision	6		6				
Application provision	41	4	19	1	21	1	3
Enactment date	18		16				2
Short title	4		4				
Change – Scope	55	5	60				
Change – Insertion	44		42	2			
Change – Replacement	111		102	7			2
Change – Repeal	23		18	4			1
Change – Renumbering	9		9				
	330	13	294	14	21	1	13

Table 17: Expected results for modelling sentences based on patterns

Though most of the sentences could be handled with the patterns as described in section 5.1, this did not automatically guarantee a correct outcome. Within the sentences that did conform to the patterns, we noted two problems that might lead to problems. First of all, several of the application provisions have conditions that need to be parsed in order to be properly modelled. So, despite the main sentence conforming to the pattern, the sentence as a whole might not be modelled correctly. In addition, some sentences contained references to renumbered or newly inserted elements, such as *sub 4 (new)* or *item b that has been renumber to item c*. Some other sentences refer to the header of an element, such as *the header of section 5*. As the reference parser as described in section 3.4 does not yet (completely and fully) detect such references, these sentences would not yet be correctly modelled.

As the table shows, the corpus contains 23 delegations. Eighteen of these follow a standard pattern. Three sentences did not follow the standard pattern of defining a topic on which rules may be created, but rather defined actions that could be taken, thus deviating from the standard pattern. Another one referred back to an earlier provision to describe the topic of the rules allowed, using a non-standard pattern. The last sentence split the topic of those rules in two parts, also deviating from the standard patterns:

Bill 31 726 nr. 2, article I, sub P, quoted text (to replace text in the Base Databases Act)¹⁴³

By order-in-council, to protect the private life of persons to whom data that has been included in the base databases, as meant in article 2, can be traced, restrictions may be set for specific data or categories of data with regards to the stipulations in sub 1.

The six publication provisions in the corpus all follow the basic pattern. There are 45 application provisions in the corpus. Of these 45, nineteen follow the most basic format, having no conditions. Fourteen sentences include a topic on which the target does or does not apply (following the word *on* or the words *with regards to*), four sentences include an explicit condition (an auxiliary *if*-sentence), and one sentence has both.

For these sentences, the application provision itself conforms to the patterns, and can be decomposed into the different frame elements. However, the topic and conditions will also need to be modelled, and that can be a more complicated task (involving the methods discussed in section 5.3). The next provision shows how complicated these topics may become:

Bill 31 531 nr. 2, article I, quoted text (to replace a text in the Competition Act)¹⁴⁴

Without prejudice to sub 1, article 6, sub 1, does also not apply to contracts, decisions and collusive behaviour as meant in that article, insofar enterprises or associations of entrepreneurs are involved who are actual or potential competitors on one or more relevant markets, if the combined market shares of the enterprises or associations of entrepreneurs involved in the contract, decision or collusive behaviour does not exceed 10% on any of the relevant markets to which the contract, decision or collusive behaviour pertains.

¹⁴³ **Wetsvoorstel 31 726, artikel I, lid P, geciteerde tekst (ter vervanging van een tekst in de Wet basisregistraties)**

Bij algemene maatregel van bestuur kunnen ter bescherming van de persoonlijke levenssfeer van personen tot wie de gegevens die zijn opgenomen in de basisregistraties, bedoeld in artikel 2, herleidbaar zijn voor daarbij aangewezen gegevens of categorieën van gegevens beperkingen worden vastgesteld ten aanzien van het bepaalde in het eerste lid.

¹⁴⁴ **Wetsvoorstel 31 531 nr. 2, artikel I, geciteerde tekst (ter vervanging van een tekst in de Mededingingswet)**

Onverminderd het bepaalde in het eerste lid, geldt artikel 6, eerste lid, voorts niet voor overeenkomsten, besluiten en onderling afgestemde feitelijke gedragingen als bedoeld in dat artikel voor zover daarbij ondernemingen of ondernemersverenigingen betrokken zijn die daadwerkelijke of potentiële concurrenten zijn op een of meer van de relevante markten, indien het gezamenlijke marktaandeel van de bij de overeenkomst, het besluit of de onderling afgestemde feitelijke gedraging betrokken ondernemingen of ondernemersverenigingen op geen van de relevante markten waarop de overeenkomst, het besluit of de onderling afgestemde feitelijke gedraging van invloed is, groter is dan 10%.

There are two sentences that include restrictions on how the targets should apply, using the phrase *with the understanding that*. They introduce several provisions that are only relevant for the specific situation described in the application provision:

Bill 31 835 nr. 2, article 5, sub 1¹⁴⁵

The articles 9, 11 to 14, 18, 20, 21, 23, 29, 39, sub 1, and 41 of the General Act apply to the board of advisors, with the understanding that:

1. by remuneration or compensation as meant in article 14 of the General Act is understood: holiday allowance; and
2. the board of advisors may issue the annual report as meant in article 18, sub 1, of that law together with the annual report of the management.

Though such sentences do follow a regular format (and the additional provisions follow a regular format as well), the current frame does not accommodate these features.

Two sentences describe the conditions in which their target applies by referring to the application of other sections of the law:

Bill 30 411 nr. 2, article 3.2, sub 4¹⁴⁶

With regards to the application of sub 2, the Financial Services Act as well as the articles 2.7, sub 3, 2.15f and 2.16 of this law do apply.

One sentence referred to a modified target, which is not yet recognised by the reference parser. Three sentences do not use an explicit reference for the target, referring to the law in general, and in that deviate from the standard pattern:

Bill 30 435 nr. 2, article III, sub 2¹⁴⁷

With regards to the untimely decision on a request or appeal which has been lodged before the time of enactment of this law, the law as it applied before that time remains applicable.

Of the eighteen enactment date provisions in the corpus, sixteen follow the generic patterns. One of the deviating sentences includes an (also non-standard) application provision as an auxiliary sentence:

¹⁴⁵ **Wetsvoorstel 31 835 nr. 2, artikel 5, eerste lid**

De artikelen 9, 11 tot en met 14, 18, 20, 21, 23, 39, eerste lid, en 41 van de Kaderwet zijn van overeenkomstige toepassing op de raad van advies, met dien verstande dat:

- a. onder bezoldiging of schadeloosstelling als bedoeld in artikel 14 van de Kaderwet wordt verstaan: vacatiegeld; en
- b. de raad van advies het jaarverslag, bedoeld in artikel 18, eerste lid, van die wet tezamen met het jaarverslag van het bestuur kan uitbrengen.

¹⁴⁶ **Wetsvoorstel 30 411 nr. 2, artikel 3.2, vierde lid**

Met betrekking tot de toepassing van het tweede lid zijn de Wet financiële dienstverlening alsmede de artikelen 2.7, derde lid, 2.15f en 2.16 van deze wet van overeenkomstige toepassing.

¹⁴⁷ **Wetsvoorstel 30 435 nr. 2, artikel III, tweede lid**

Ten aanzien van het niet tijdig beslissen op een aanvraag die of een bezwaarschrift dat is ingediend voor het tijdstip van inwerkingtreding van deze wet, blijft het recht zoals dit gold voor dat tijdstip van toepassing.

Extraordinary Decree Labour Relations 1945, article 33, sub 1¹⁴⁸

This decree, with regards to which the competence, meant in article 9, sub 2, of the Decree on the Extraordinary State of Emergency cannot be exercised, is enacted starting on October 15th, 1945.

The other also includes an application provision, and is actually a mixed type sentence, consisting of two sentences that are concatenated:

Bill 30 583 nr. A, article II¹⁴⁹

This law is enacted starting on the day after the date of publication of the Bulletin of Acts and Decrees in which she is included, and applies to the time periods that start after the enactment date of this law.

The four short title provisions in the corpus all follow the generic pattern, as do the 60 scope provisions. Of the 44 insertion provisions, 38 follow the basic pattern, while six others follow a more complicated pattern: four of these provisions also renumber some items, one provision makes multiple insertions (the instruction includes *each time* to indicate that the insertion has to be repeated), and one has an auxiliary sentence to prevent a clash with another change.

There are 111 replacement provisions, of which 45 follow the basic pattern for the replacement of a structure element, including one sentence that replaces a header. There is one replacement of a structure element that also renumbers, and one that contains an auxiliary sentence to prevent a clash with another change.

There are 49 replacement provisions that follow the basic pattern for the replacement of text. Seven make multiple replacements using *each time*, and six makes multiple replacements by only specifying multiple targets. One of the multiple replacements also changes a header.

Two replacements deviate from the patterns. One of the text replacements does not specify which text must be replaced by quoting it, but describes it as *the definition of 'Our Minister'*:

Bill 22 139 nr. 2, article I, sub A¹⁵⁰

In article 1, sub 1, the definition of 'Our Minister' is replaced by: Our Minister of Agriculture, Nature and Fishing.

Another deviates by using two references rather than one to define the target of the replacement:

¹⁴⁸ **Buitengewoon Besluit Arbeidsverhoudingen 1945**

Dit besluit, ten aanzien waarvan de bevoegdheid, bedoeld in artikel 9, tweede lid, van het Besluit op den bijzonderen staat van beleg niet kan worden uitgeoefend, treedt in werking met ingang van 15 oktober 1945.

¹⁴⁹ **Wetsvoorstel 30 583 nr. A, artikel II**

Deze wet treedt in werking met ingang van de dag na de datum van uitgifte van het Staatsblad waarin zij wordt geplaatst en is van toepassing op tijdvakken die aanvangen na inwerkingtreding van deze wet.

¹⁵⁰ **Wetsvoorstel 22 139 nr. 2, artikel I, lid A**

In artikel 1, eerste lid, wordt de begripsomschrijving van 'Onze Minister' vervangen door: Onze Minister van Landbouw, Natuurbeheer en Visserij.

Bill 22 139 nr. 2, article I, sub E¹⁵¹

In article 15, the subparagraphs 1, 2 and 3 will read: ...

In this sentence, *article 15* and *subparagraphs 1, 2 and 3* are two separate references rather than the single reference expected (this is more obvious in the original Dutch sentence, where a verb appears between the two references).

There are 23 repeals. Five of these follow the basic pattern for repealing an entire structure element, and eight follow the basic pattern for repealing part of the text of an element. Four sentences modify the header of a text (thus affecting the index as well, and effectively renumbering the element). Two sentences make multiple repeals (using *each time*, as with the insertion provisions) and three sentences also renumber some items. One repeal does not follow a standard pattern, as it includes a reference to the date of the repeal:

Extraordinary Decree Labour Relations 1945, article 33, sub 2¹⁵²

From that date, the Extraordinary Decree Labour Relations (Decree of July 17th, 1944, no. E 52, last modified by decree of December 29th, 1944, no. E 157) is repealed.

There are nine (pure) renumbering provisions in the corpus; six of those are regular renumberings, and the other three are sentences that add an index to a paragraph that did not have one before. These are both standard patterns.

All in all, a large number (330 out of 343, or 96%) of the sentences can be modelled correctly using the rules as presented. However, this does not yet guarantee perfect results, as two more issues are involved. First of all, the reference parser needs to identify all references, and currently, it does not yet recognise references to renumbered articles, which are used in some of the provisions. Secondly, delegations and application provisions include large portions of domain-specific text, which needs to be modelled further.

For this type of sentences, the pattern-based classifier manages about 92% accuracy (the machine-learning classifier achieves 97%). If we assume that there are errors made in classification are independent of those made in modelling, then this suggests that about 88% (96% of 92%) of these sentences can be correctly classified and modelled. As these sentences comprise about 57% of our corpus, this comes down to about 50% of the entire corpus.

5.3 Sentences Dealing With Subject Matter

Sentences dealing with the subject matter of the law come in many shapes and varieties, and though there are some clear signal words, it is difficult to extract the features of a sentence based upon those signal words. However, a sentence contains more information than just signal words. Sentence structure, grammatical conjugation of verbs, etc., can help us determine the different elements from a sentence and their roles. Thus, when processing sentences that deal with the subject matter of the law, we will first parse the sentence, using the Alpino

¹⁵¹ **Wetsvoorstel 22 139 nr. 2, artikel I, lid E**

In artikel 15 worden het eerste, tweede en derde lid vervangen door: ...

¹⁵² **Buitengewoon Besluit Arbeidsverhoudingen 1945, artikel 33, tweede lid**

Dit besluit, ten aanzien waarvan de bevoegdheid, bedoeld in artikel 9, tweede lid, van het Besluit op den bijzonderen staat van beleg niet kan worden uitgeoefend, treedt in werking met ingang van 15 oktober 1945.

parser, created by Bouma, van Noord and Malouf (2001). This is a parser for the Dutch language, which has been evaluated and tested using sentences taken from newspaper articles. It assigns a dependency structure to a sentence. These structures are described by Bouma et al.:

Dependency structures make explicit the dependency relations between constituents in a sentence. Each non-terminal node in a dependency structure consists of a head-daughter and a list of non-head daughters, whose dependency relation to the head is marked.

We will then use the output of the parser (a parse tree, which is stored in XML) as the input for our modelling process.

We will first discuss the general approach for creating a model of a normative sentence or definition from a parse tree. After that, several additional considerations are discussed. We will not discuss calculations. Those have a great variety in different patterns that each require a separate model, and which may be nested as well. Discussing those would require too much effort, as we wish to focus on the (more common) normative sentences.

5.3.1 Normative Sentences

We see each normative sentence as describing a situation that is allowed or disallowed. We consider the main verb of a sentence as the action that is allowed or disallowed, with the other elements being modifiers or properties of that action. A number of these other elements are labelled according to their semantic role (or thematic relation) in the sentence. The semantic role describes the role a participant plays in a certain situation, which may differ from the role it has in the sentence (such as subject or object). An example taken from Payne (1997):

If, in some real or imagined situation, someone named John purposely hits someone named Bill, then John is the agent and Bill is the patient of the hitting event. Therefore, the semantic role of Bill is the same (patient) in both of the following sentences:

- John hit Bill.
- Bill was hit by John.

In both of the above sentences, John has the semantic role of agent.

At the moment, we distinguish only the agent, patient and recipient of the action. The other elements are considered as generic modifiers. Other researchers have already been working at classifiers to assign semantic roles (Stevens et al, 2006), and we hope to adopt one of those in the future, but for the moment, we use two simple schemes for labelling them, one for active sentences, and one for passive sentences.

In an active sentence, we assume that:

- the subject is the agent of the action;
- the direct object is the patient of the action;
- the indirect object is the recipient of the action.

For example:

Our Minister issues a warrant to the negligent person.

The main verb of this sentence is *to issue*, so that is considered the action. Properties of this action are the agent (*Our Minister*), the patient (*a warrant*) and the recipient (*the negligent person*). All these elements are distinguished by the Alpino parser (as subject, direct object and indirect object), allowing us to extract them for our model.

Within Dutch law, this sentence format expresses an obligation, so the action as a whole is classified as an obligation.

Obligation	
Action	issue
Agent	Our Minister
Patient	warrant
Recipient	negligent person

The articles (*the, a*) are left out of the model, though they are stored internally, as they are of importance during later integration of the model; *the negligent person* often is a reference to an earlier sentence, whereas *a negligent person* is not.

The example above is an active sentence, but many sentences in Dutch law are phrased in the passive voice, such as this instruction:

Protection of Antarctica Act, article 33, sub 3, second sentence¹⁵³

An English translation is added to this report.

A sentence in the passive voice cannot be modelled in the same way as a regular sentence, as the subject of the sentence is not the agent, but the patient, and should be modelled as such. The parse of the sentence gives us an easy way to do this:

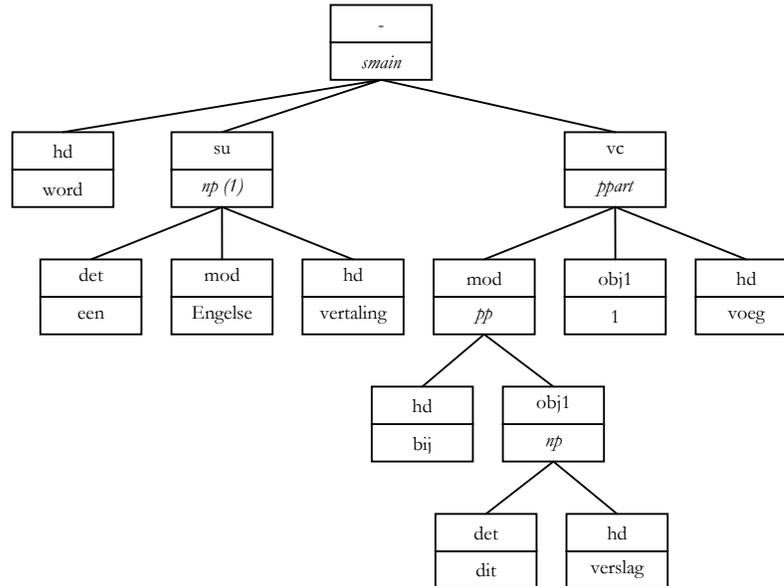


Figure 14: Alpino parse tree (with reduced information) for *An English translation is added to this report* (in Dutch).

¹⁵³ **Wet bescherming Antarctica, artikel 33, derde lid, tweede volzin**
Bij dit verslag wordt een Engelse vertaling gevoegd.

The verb clause (vc) of the sentence holds the sentence in active voice, with the subject re-cast in the role of object. This re-cast subject is our cue that we are dealing with a passive sentence. By modelling the verb clause instead of the sentence as a whole, we get the correct model, with the correct object, and without the auxiliary verb. So, we will the frame based on the nodes that appear under the verb clause. As there is no subject listed there, there will be no agent in the frame. There is an object present (which refers to the subject of the sentence as a whole), so we will add a patient to the frame.

If the agent is present in the sentence (for example, if the sentence would read *An English translation is added to this report by the organiser*), then this prepositional object is not re-cast in the role of object in the tree. We will have to detect its presence by scanning for signal words like *by*. As this does not always indicate the agent, this will be one of the cases where human validation is necessary. Further detail can be added by splitting of adjectives and relative clauses from the noun they modify. For example, *negligent person* has two properties: being a person and being negligent. Splitting adjectives from nouns is not always desirable; it is preferable to leave multiword expressions intact. *European Union* is not any union that is also European; *Our Minister of Finance* is not any minister that is also ours, and of finance¹⁵⁴. Instead, these are references to concepts that have been defined elsewhere: the common sense domain, the legal domain or elsewhere in this law. Common multiword expressions are recognised by the Alpino parser; legal domain or law-dependent expressions need be filtered out separately.

Relative clauses are more complex than adjectives, as they contain a complete new sentence. In this case, we repeat the procedure for the main sentence, identifying the main action and all properties of that action. For example:

Our Minister issues a warrant to the person that neglected his duties.

This sentence yields a frame like:

Obligation							
Action	issue						
Agent	Our Minister						
Patient	warrant						
Recipient	person						
	<table border="1"> <thead> <tr> <th colspan="2">AgentOf</th> </tr> </thead> <tbody> <tr> <td>Action</td> <td>neglect</td> </tr> <tr> <td>Direct Object</td> <td>his duties</td> </tr> </tbody> </table>	AgentOf		Action	neglect	Direct Object	his duties
AgentOf							
Action	neglect						
Direct Object	his duties						

5.3.1.1 Filtering Out Signal Words

The sentences we showed above are examples of normative sentences that do not use signal words; only the desired situation is described, and it is left implicit that this is an obligation. Other sentences in the law use signal words to make the kind of norm explicit, such as:

¹⁵⁴ In Dutch laws, *Our Minister of Finance* is a reference to the (Dutch) Minister of Finance. No more detailed model is needed, as no derivations need to be made.

Civil Code, book 7, article 26, sub 1¹⁵⁵

The buyer is obliged to pay the price.

This sentence uses *is obliged* to make it clear that this is an obligation. Other examples of signal words are *must*, *may* and *is allowed*. These sentences require a different approach than the sentences without signal words. If we were to use the same approach, the result would be something like:

Obligation	
Action	is obliged to pay
Agent	buyer
Patient	the price

This is not a desirable outcome, as the action that this norm deals with is *pay* rather than *is obliged to pay*. When modelling these sentences, these signal words should not be included in the model of the situation (their meaning is translated into whether the situation is allowed or disallowed). Ideally, after we have categorised the sentence (based on the signal words), we would like to transform the sentence to a sentence without signal words, like:

The buyer pays the price.

We could then model that sentence to come to a correct frame. Simply leaving out the signal words may lead to errors, since the role of the other words might need to shift as well. However, the parse of the sentence actually contains this “transformed sentence” that we want to model. This is shown in figure 15.

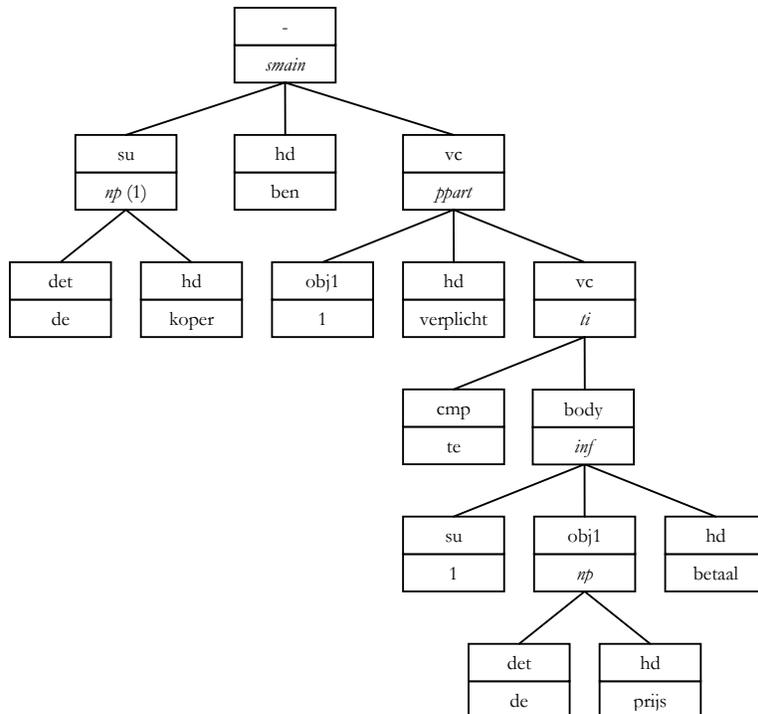


Figure 15: Alpino parse tree (with reduced information) for *The buyer is obliged to pay the price* (in Dutch).

¹⁵⁵ **Burgelijk Wetboek Boek 7, artikel 26, eerste lid**

De koper is verplicht de prijs te betalen.

Beneath the body node, we find exactly the “sentence” that we are looking for. Alpino assigns this dependency structure to any sentence that follows this pattern. This makes it easy to filter out the signal words by simply focusing on the part of the parse tree that contains the transformed sentence. Similar patterns, such as *is allowed to* result in a similar tree, with the actual action described beneath the body node.

For sentences that use a modal auxiliary verb, we can also focus on a specific sub-tree. For example, the sentence:

The buyer must pay the price.

The parse tree for this sentence is shown in figure 16.

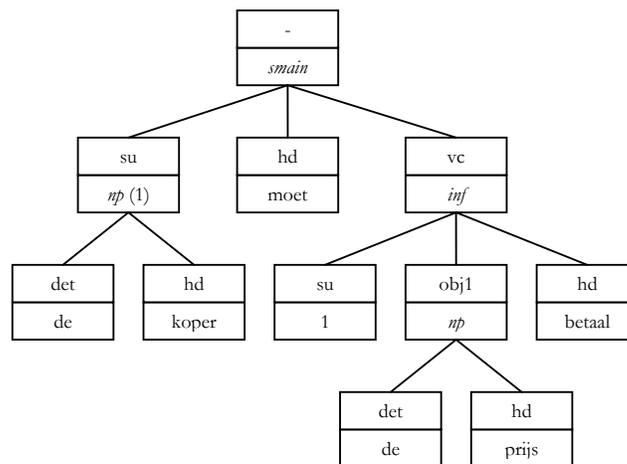


Figure 16: Alpino parse tree (with reduced information) for *The buyer must pay the price* (in Dutch).

Again, there is a sub-tree (this time under the *vc* node) that corresponds to the action that we want to represent in the model (i.e. *the buyer pays the price*). For other modal auxiliary verbs, the parse tree is similar.

So, depending on the pattern used, we use a different sub-tree as the basis for our frame:

- no pattern used (normative indicative): use the entire tree;
- modal auxiliary verbs (*must, may, can*): use the sub-tree under the (top-most) *vc* node;
- a pattern *is ... to* (*is obliged to, is permitted to, is allowed to*): use the sub-tree under the (top-most) *ti* node.

5.3.1.2 Lists

Lists are also recognised by the Alpino parser, and can therefore easily be added to our models as the union or intersection of the different list items, depending on the conjunction used. However, though the conjunction *and* suggests an intersection, it often expresses a union instead. For example:

Advances and duties are paid in cash.

In this sentence, it is the union of *advances* and *duties* that is meant. Our current approach is to translate *and* with a union if it appears in a relative clause, and with an intersection otherwise.

5.3.1.3 Negation

It is important to recognise negations such as *no* or *not*. For example:

No bodies are interred on a closed cemetery.

This is an obligation, and the patient of this sentence is *no bodies*. So, this sentence would lead to the following frame.

Obligation	
Action	inter
Patient	no bodies
Condition	on a closed cemetery

This frame implies that one is obliged to bury something that is not a body on a closed cemetery, which is not what is intended in the law. The actual meaning is that one should not bury bodies on a closed cemetery, represented as:

Obligation	
Not	
Action	inter
Patient	bodies
Condition	on a closed cemetery

If we encounter a negation in a (main) sentence, and it is not part of the pattern used to classify the sentence, then this leads to a negation of the frame (using the not operator), and the word *not* or *no* is not included as text in the frame. If the negation was part of the pattern used to classify this sentence, its meaning is already represented in the type of the frame, and the negation does not lead to a negation of the frame, nor is it included in the text.

If we encounter such a negation in an auxiliary sentence, then this leads to the negation of the frame representing that auxiliary sentence, not to negation of the frame for the main sentence.

5.3.1.4 Explicit Exceptions

Sometimes, a normative sentence in a Dutch law includes a prefix to denote that it is an exception to some other rule, like:

In exception to article 12, ...

Alternatively, some sentence start with a prefix to denote that it is not an exception, like:

Without prejudice to article 12, ...

These prefixes differ from other elements in the sentence in the sense that they do not describe the situation that is allowed or obliged, but instead tell us something about how this rule interacts with some other rule. Hence, this element should not be added to the frame describing the rule. Instead, it becomes a relation between this sentence and the referenced structure element.

5.3.1.5 Conditions

Auxiliary sentences that express a condition to the main sentence need to be modelled separately. The different classes of conditions and their representations have not been studied in detail as part of this research, so the descriptions given here are incomplete.

Just as there are sentences dealing with the law and sentences that deal with the subject matter of the law, there are also conditions dealing with the law and conditions that deal with the subject matter of the law.

Like a normative sentence, a condition that deals with the subject matter of the law specifies a situation, and it can be modelled in a similar manner. The result is a condition frame describing a situation, which is linked to the frame representing the main sentence (which usually represents a norm or an application provision). For example:

Coin Act 2002, article 8, sub 3¹⁵⁶

If the coins are counterfeits, as judged by the office meant in sub 2, they are returned cut in half or the raw material value is reimbursed.

Here, the main sentence is an obligation: *they are returned cut in half or the raw material value is reimbursed*. The auxiliary sentence forms a condition, as is represented as a separate frame linked to the obligation (alternatively, it could also be represented as a nested frame, which has the same meaning). The result is:

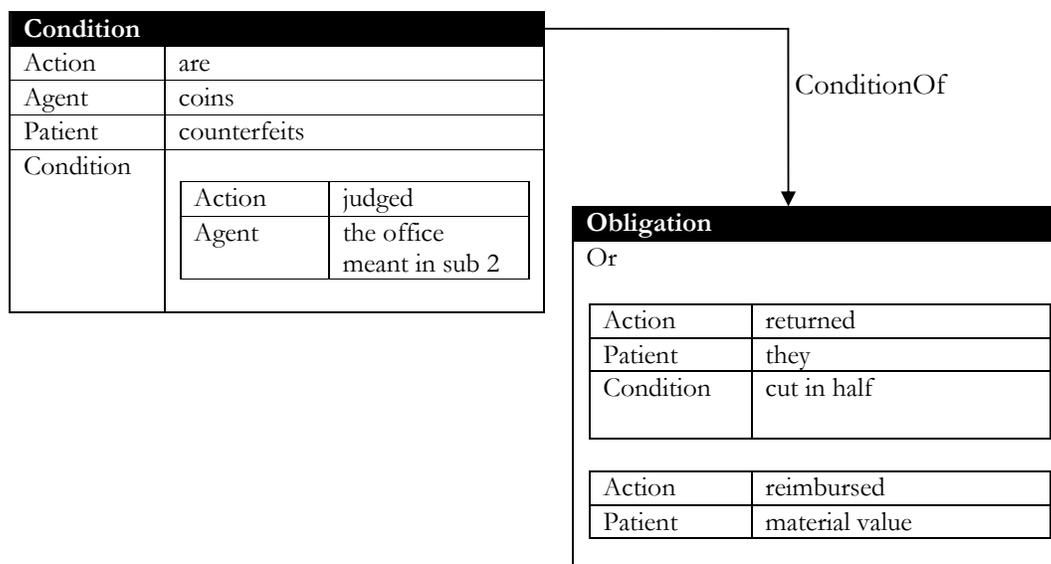


Figure 17: Linked condition and obligation frame

An important difference with normative sentences is that conditions may specify situations that lie in the past. This means that when creating the frame for a condition, the tense of the

¹⁵⁶ Muntwet 2002, artikel 8, derde lid

Indien de munten naar het oordeel van de in het tweede lid bedoelde instantie vals of vervalst zijn, worden ze doorgesneden teruggegeven of wordt de stoffelijke waarde vergoed.

sentence (i.e. present tense, past tense etc.) needs to be preserved. Also, some situations do not involve an action, but rather describe some property or attribute of involved items, such as the example given above. In this case, rather than modelling this as an action based on the verb *to be*, it may be more desirable to translate such conditions using *subclassOf* relations.

The conditions that deal with the law refer to the applying or applicability of other parts of the law, such as:

Prosecution Code, article 366a, sub 1¹⁵⁷

In case article 14a or 77x of the Penal Code has been applied, ...

Or:

Work and Income to Working Capacity Act, article 64, sub 5¹⁵⁸

If sub 4 applies, ...

Like application provisions, such conditions should be modelled using dedicated frames rather than the frames used for generic situations in the application domain.

5.3.2 Definitions and Deeming Provisions

Definitions and deeming provisions attach a meaning to a specific concept. At top level, a definition contains three elements: the *definiendum* and the *definiens*, and, optionally, a scope declaration stating for which sources of law this definition applies. For example:

Medication Act, article 1, introduction and item c¹⁵⁹

In this law and stipulations based upon it, it is understood by immunological drug: a vaccine, toxin, serum or allergen.

This definition has the scope this law and stipulations based upon it. The definiendum is immunological drug and the definiens is a vaccine, toxin, serum or allergen. The parse tree for the definiens is shown in figure 18.

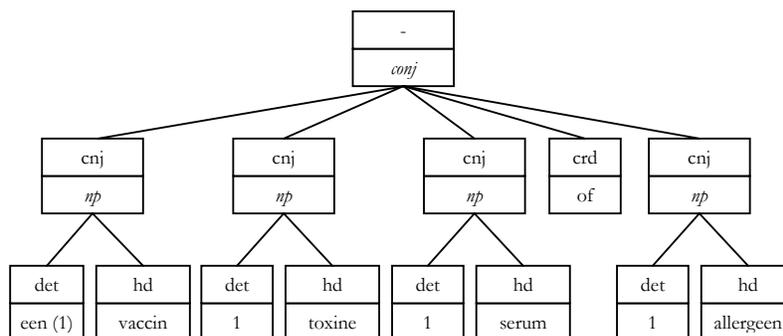


Figure 18: Alpino parse tree (with reduced information) for *a vaccine, toxin, serum or allergen* (in Dutch).

¹⁵⁷ **Wetboek van strafvordering, artikel 366a, eerste lid**

In geval artikel 14a of 77x van het Wetboek van Strafrecht is toegepast, ...

¹⁵⁸ **Wet arbeid en inkomen naar vermogen, artikel 64, vijfde lid**

Indien het vierde lid van toepassing is, ...

¹⁵⁹ **Geneesmiddelenwet, artikel 1, aanhef en onderdeel c**

In deze wet en de daarop berustende bepalingen wordt verstaan onder immunologisch geneesmiddel: een vaccin, toxine, serum of allergen;

Like the sentences presented in section 2, these top elements can easily be extracted by means of the pattern used to classify the sentence (in this case it is understood by) and some additional features. The scope, if present, will follow the word in (and end at the text it is understood by). The definiens will follow the word by and end at the colon, and the definiendum will follow the colon¹⁶⁰. Thus, we can easily extract a top level frame:

Definition	
Scope	This law and stipulations based upon it
Definiendum	immunological drug
Definiens	a vaccine, toxin, serum or allergen

However, modeling the definiens in this way is unsatisfactory, as it gives insufficient detail to use this model for practical purposes. To create a more useful model, we need to split up the definiens. To do so, we use the same methods used for concepts in normative sentences. This requires a parse tree; we can either parse the entire sentence or just the definiens.

5.4 Example: Converting Frames to OWL

The frames presented in this chapter do not form an executable specification. However, they contain sufficient information to be converted to a different notation that can be executed. In this section, we will show how the frame we have created for normative sentences can be converted into OWL expressions, which can be evaluated using the HARNES system (van de Ven et al., 2008).

In HARNES, a norm is represented in terms of the LKIF Core ontology (Hoekstra et al., 2007) as a deontic qualification of a generic case (Valente, 1995). Such a generic case is a conjunction of conditions that together form a description of the situation expressed by the norm. It is defined as a set of conditions in conjunctive normal form. An individual case is a set of grounded propositions that describe a certain state of affairs.

The norm itself is qualified using the deontic notions Permission, Obligation and Prohibition, which have been defined in the LKIF Core ontology. A norm is a qualification, which normatively qualifies something (as opposed to non-normative qualifications)¹⁶¹:

```
Class: Norm
SubClassOf:
    Qualification and qualifies some Normatively_Qualified
```

A permission is a norm that allows some situation:

```
Class: Permission
SubClassOf:
    Norm
EquivalentTo:
    allows some Allowed and allows only Allowed
```

¹⁶⁰ There are a number of different patterns, but for each pattern, a similar set of features can be identified to extract these three elements.

¹⁶¹ The OWL statements have been written in the Manchester notation (see Horridge et al., 2006).

An obligation is something that allows something (i.e. that what is obliged) and also disallows something (i.e. the situation in which that what is obliged is not performed).

```
Class: Obligation
SubClassOf:
    Permission
EquivalentTo:
    allows some Obligated and disallows some Disallowed
    and allows only Allowed and disallows only Disallowed
```

A prohibition is equivalent to an obligation, as a prohibition to do something is the same as the obligation not to do it:

```
Class: Prohibition
EquivalentTo:
    Obligation
```

In an example, van de Ven et al. (2008) describe how a set of library rules can be represented using HARNESS. These rules apply to the checking out of books. The example starts with specifying a default situation. In this case, the default situation is disallowed: students are not allowed to check out books, except when the norms state something else. This default norm and generic case apply to all situations where a book is checked out.

```
Class: Default_CG
SubClassOf:
    Generic_Case and disallowed_by some {defaultnorm}
EquivalentTo:
    checks_out some Library_Book
```

```
Class: Default_Norm
SubClassOf:
    Prohibition and disallows only Default_GC
EquivalentTo:
    {defaultnorm}
```

These statements say that the default generic case is a generic case, which is disallowed by the default norm. The situations to which it applies are those situations in which a library book is checked out. The default norm is a prohibition which disallows the generic case.

Other norms are represented in a similar manner as the default norm. The example continues with article 1a of the regulations, which states: *Students registered at this university are allowed to check out a book from this library.* This is represented as the Article 1a Generic Case, which is a registered student checking out a library book. This situation is permitted by article 1a.

```
Class: Art1a_CG
SubClassOf:
    Generic_Case and allowed_by some {art1a}
EquivalentTo:
    Registered_Student and checks_out some Library_Book
```

```

Class: Art1a_Permission
SubClassOf:
    Permission and allows only Art1a_GC
EquivalentTo:
    {art1a}

```

These representations can now be used to assist in the task of legal assessment. This is done by specifying an individual case, such as this:

```

{Amy: Registered_Student, general_book_1: Library_Book, general_book_2:
Library_Book, general_book_3: Library_Book, general_book_4: Library_Book,
general_book_5: Library_Book, general_book_6: Library_Book, Amy
checks_out general_book_1, Amy checks_out general_book_2, Amy checks_out
general_book_3, Amy checks_out general_book_4, Amy checks_out
general_book_5, Amy checks_out general_book_6 }

```

Using an OWL classifier, this individual case can be classified as belonging to any of the generic cases. The case of Amy, above, matches both the default generic case (as there are books being checked out) as the Article 1a generic case (as it is a registered student checking out books). Based on the norms allowed to these generic cases, it is possible to determine whether this individual case is allowed. This situation is allowed by Article 1a (as it matches the Article 1a generic case), but disallowed by the default norm (as it matches the default generic case). However, any explicit norm is considered an exception to the default norm, and has precedence, leading to the conclusion that this individual case is allowed¹⁶².

So, in order to add a normative sentence to this system, we need to specify the generic case, as a set of conditions. We continue with the example from section 5.3.1:

Our Minister issues a warrant to the negligent person.

After parsing, this sentence yielded the following frame:

Obligation	
Action	issue
Agent	Our Minister
Patient	warrant
Recipient	negligent person

We can describe the generic case using the elements from this frame. The generic case is an action *issue* with agent *Our Minister*, patient *warrant* and recipient *negligent person*.

```

Class: Generic_Case
EquivalentTo:
    issue and agent some Our_Minister and patient some warrant
and recipient some negligent_person

```

This generic case is allowed by the article, and its negation is disallowed by the article, which leads to the following complete statement in HARNESS:

¹⁶² This exception relation is derived outside OWL.

```

Class: Generic_Case_P_Example1
SubclassOf:
    Generic_Case and allowed_by some {example1}
EquivalentTo:
    issue and agent some Our_Minister and patient some warrant
    and recipient some negligent_person

```

```

Class: Generic_Case_F_Example1
SubclassOf:
    Generic_Case and disallowed_by some {example1}
EquivalentTo:
    not Generic_Case_P_Example1

```

```

Class: Example1_Obligation
SubclassOf:
    Obligation and allows only Generic_Case_P_Example1
    and disallows only Generic_Case_F_Example1
EquivalentTo:
    {example1}

```

The example *The buyer is obliged to pay the price* results in the frame:

Obligation	
Action	Pay
Agent	buyer
Recipient	Price

This translates to the following HARNESS statement:

```

Class: Generic_Case_P_Example2
SubClassOf:
    Generic_Case and allowed_by some {example2}
EquivalentTo:
    pay and agent some buyer
    and patient some price

Class: Generic_Case_F_Example2
SubclassOf:
    Generic_Case and disallowed_by some {example2}
EquivalentTo:
    not Generic_Case_P_Example2

Class: Example2_Obligation
SubclassOf:
    Obligation and allows only Generic_Case_P_Example2
    and disallows only Generic_Case_F_Example2
EquivalentTo:
    {example2}

```

A sentence denoting a permission, such as *A holding cell may be equipped with an observation camera* is treated in a similar manner. The frame for this sentence is:

Permission	
Action	equip
Patient	holding cell
Prepositional Object ¹⁶³	observation camera

The translation for this frame again follows the same approach again, with the difference that this sentence only includes a permission, and not a prohibition:

```

Class: Generic_Case_P_Example3
SubClassOf:
    Generic_Case and allowed_by some {example3}
EquivalentTo:
    equip and patient some isolation_cell
    and pobject some observation_camera

Class: Example3_Permission
SubClassOf:
    Permission and allows only Generic_Case_P_Example3
EquivalentTo:
    {example3}

```

As HARNES is modelled in OWL, any definitions can be easily modelled as well, by adding an equivalent relation between the definiens and the definiendum. For example, the definition given in section 5.3.2:

In this law and stipulations based upon it, it is understood by immunological drug: a vaccine, toxin, serum or allergen.

Becomes the following OWL statement:

```

Class: immunological_drug
EquivalentTo:
    vaccine or toxin or serum or allergen

```

This OWL statement does not contain a translation for the scope declaration *In this law and stipulations based upon it*. Within HARNES, no method has been defined for dealing with scope declarations, and OWL has no obvious mechanisms to deal with this.

One way to deal with this is to simply define each term with the same label separately – a *common_sense_immunological_drug*, a *medicine_law_immunological_drug*, etc. This approach means that each partial model has to be integrated by hand with the correct definition; a human expert will need to determine which definition to use, as the knowledge of the scope of the definition is not modelled. When a new definition is added – suppose the Medicine Law is amended to include a broadened definition of immunological drug that applies to chapter 7 only, thus introducing the concept *medicine_law_chapter7_immunological_drug* – then all rules that use this definition must be updated by hand.

¹⁶³ “observation camera” does not fill one of the semantic roles that we currently assign, and hence is described with the more primitive label “Prepositional Object”.

A more promising method might be to develop an approach similar to that of Klarman, Hoekstra and Bron (2008) who describe a method for concepts that have multiple versions in time.

5.5 Experiences

At this moment, we do not have a fully automated process to create the models, and have not yet tested this method on a large body of sentences. Instead, random sentences have been selected, parsed using Alpino and then fed into our modeller.

There is a clear difference between the computer generated models and those created by a human expert with regard to the granularity of the model. Our method will create models with model elements that represent one word from the original sentence, whereas a human expert is more likely to include some sentence fragments as a whole. For example, one Dutch law defines an alcoholic drink as *the drink that, at a temperature of twenty degrees Celsius, consists of alcohol for fifteen or more volume per cents, with the exception of wine*. Our algorithm will split this sentence into many elements, whereas most human modellers will leave the first subordinate sentence intact and add it to the model as a single attribute (most likely abbreviated to *alcohol by volume*). A more detailed model seems not necessarily wrong, but quite possibly over-the-top and inconvenient for many applications.

The method assigns rather broad categorisations to each object (it is either a direct, indirect or prepositional object), but does not yet assign a legal meaning to such an object. It may be a third party involved or the instrument. Perhaps this is not an obstacle; users dealing with a system based on such models are likely to recognise the roles from the context and language used, whereas a computer does not need this information for the derivations we currently want to make. For future projects, though, the information may be required, and some way to automatically recognise it is desired. Here, the addition of a more advanced semantic role labelling tool can be valuable.

For the modelling of norms, we have been focussing on the sentences that represent an obligation, duty or right. For those sentences, the method seems adequate. However, for other types of sentences, such as delegation, we have not come to an acceptable approach yet. Dealing with these sentences will require first of all that we recognise them. Currently, our classifier distinguishes only between obligation/prohibition and right/permission. Several of the patterns used clearly indicate delegations, but we have not yet established whether these patterns cover all delegations in Dutch laws.

A problem with regard to the parses made by Alpino is that most often, the correct parse is not the one preferred by Alpino, but second, third or fourth. If we make several suggestions (each suggestion based on a parse by Alpino), this means that it will often not be the first suggestion that is correct, which means more effort is needed by a human expert who is verifying the models. This may be caused by the fact that Alpino has been trained on newspaper articles rather than legal texts. We expect that by recalibrating the disambiguation on a written legal corpus, and perhaps by expanding the lexicon used by Alpino, this problems will disappear.

5.6 Integrating models

After a model has been generated for each individual sentence, these individual models need to be integrated. Though some provisions may stand on their own, many rules are somehow based on earlier provisions.

This integration step starts with the models of the individual provisions (sentences) that have been generated in the previous step, as described in sections 5.1 to 5.3.

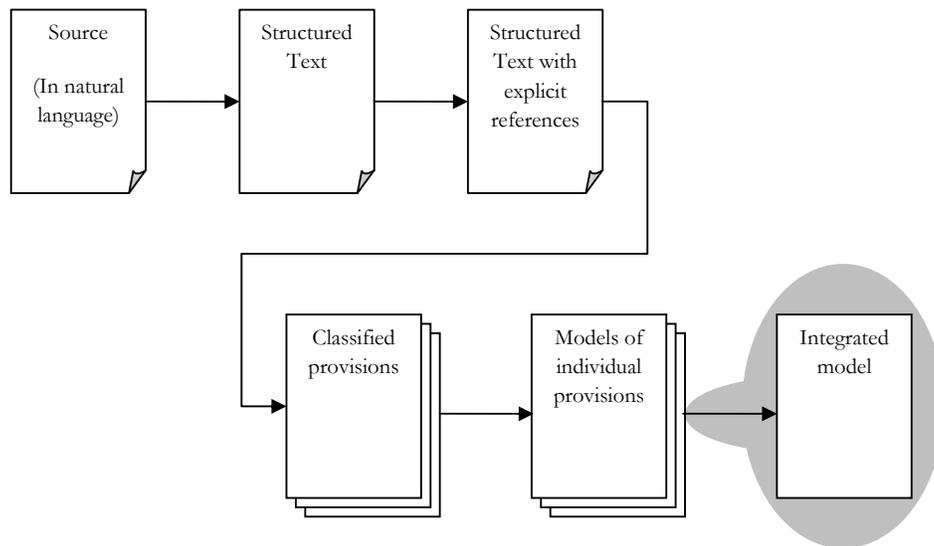


Figure 19: Integrating models step

There are several ways in which provisions can be connected. First of all, a provision can relate to a different provision (or larger element) as a whole. This is the case for modifications, application provisions, etc. as well as for certain conditions.

Provisions can also be linked through the concepts they use. The provisions that deal with the subject matter of the law will contain texts (usually a noun phrase or verb phrase) that refer to certain concepts. When creating a model for the individual sentences, each model will contain a class for that concept. For example, in a law about child subsidy, we may have *child* appear in article 5, article 6, first sentence, article 6, second sentence and article 7. When modelling these sentences separately, we also get separate classes. So, we end up with classes *child according to article 5*, *child according to article 6 first sentence*, *child according to article 7 second sentence* and *child according to article 7*. Obviously, these classes are related. There are two possibilities for such connections:

- A provision contains a class that has been defined in another provision. In this case, the class in the provision is a subclass of the class in the definition.
- A provision refers to a class introduced in another provision. In this case, the two classes are equivalent.

In the next sections, the different connections are discussed, and the manner in which they can be detected.

5.6.1 Linking Provisions that are Related to Other Provisions as a Whole

There are a number of operations for which a provision needs to relate to a different provision as a whole. These operations have been described in chapter 4: modification provisions, application provisions, setting the enactment date and setting the short title. As we have identified the references in the text, have classified the sentences and have determined the role of the references in these sentences, they have effectively been integrated with the other text.

Similarly, there are a number of auxiliary sentences that relate to other provisions as a whole: auxiliary sentence setting the scope for definitions and deeming provisions, auxiliary sentences setting explicit exceptions and conditions based on the application of other provisions. Like the provisions mentioned above, these auxiliary sentences have already been linked through their target by identifying the reference and determining the role of that reference.

5.6.2 Linking Concepts to Their Definitions

Linking the concepts that are used with their definitions is a fairly straightforward step. Basically, it is the direct matching of the word, noun phrase or verb phrase representing the concept with the word, noun phrase or verb phrase that forms the definiendum of the definition. There are three minor issues involved in this matching.

First of all, the phrase as it appears in a sentence may differ from the exact phrase that was used in the definition. A noun may be plural instead of singular, a verb may be in a different tense, etc. This is a problem that is very common in natural language processing. It is usually solved using stemming or lemmatising. A stemmer attempts to find the base form (or lemma) of the word. A lemmatiser does the same, but takes in account the context in which the word appears. By applying a stemmer or lemmatiser on the words, we can compare the lemmas rather than the actual words as they appear in the sentences, thus circumventing the problem posed by the many inflections. The Alpino parser, like most parser, includes such a lemmatiser. This means that after parsing, no additional step is needed to obtain this information.

Secondly, the appropriate definition may be in a different document. This will be most common with lower regulations (i.e. not laws). Many definitions in a law have as scope *this law and the stipulations based upon it*, meaning that the definition does not only apply to the law itself, but to any legislation that is promulgated under the authority of that law as well. Thus, when integrating these norms with their definitions, then definitions from appropriate higher legislation should be included in the search.

Finally, there may be several definitions that define the word, each with a different scope. In this case, the location of the phrase should be compared to the scope of the definitions. Obviously, if the location is outside of the scope of a definition, it should not be linked to that definition. If there are multiple definitions that have the location in scope, then the definition that has the narrowest scope should be chosen. For example, the Labour and Care Act contains the following definition for *employee*:

Labour and Care Act, article 1:1, introduction and items a and b¹⁶⁴

Unless it is determined differently, for the application of this law is understood by:

- a. employer: he who has another perform labour under an employment contract under civil law or an appointment under public law;
- b. employee: the other, meant in item a.

The scope of this definition is the entire law. However, there is a different definition for section 1 of division 2 of chapter 3:

Labour and Care Act, article 3:6, sub 1, introduction and item a¹⁶⁵

For the application of this section is understood by employee: the employee meant in article 1:1, item b, with the exception of him who, based on Division 1, Section 2, of the Sickness Benefits Act, is not an employee under that act.

This second definition forms an exception to the scope of the first definition; the first definition applies throughout the law, with the exception of chapter 3, division 2, section 1 (which is the section *this section* in item a refers to), whereas the second definition only applies in that specific section¹⁶⁶. So, sentences within that section should link to the second definition, whereas sentences elsewhere in the law link to the first definition.

5.6.3 Linking Concepts That Represent the Same Class

Next to linking concepts to their definitions, concepts may also be linked between two (non-defining) sentences, indicating that those two sentences should not be applied independently, but that the second sentence applies to the same situation as the first sentence. The same may occur within a single sentence (usually between an auxiliary sentence and the main sentence). The fact that a text represents the same class as an earlier text can be indicated in a number of different ways. Firstly, the concept may be explicitly linked by means of a reference, such as *a request as meant in sub 7*. As references have been classified earlier, these links are easily identified. Secondly, a demonstrative like *this* or *that* is used. For example, in the next sentence, *that person* refers back to *a natural person*:

Mining Act, article 21, sub 5, introduction and item a¹⁶⁷

The license expires automatically: a. if the holder is a natural person, from the day after that person dies.

¹⁶⁴ **Wet arbeid en zorg, artikel 1:1, aanhef en onderdeel a**

Tenzij anders is bepaald, wordt voor de toepassing van deze wet verstaan onder werkgever:

- a. degene die een ander krachtens arbeidsovereenkomst naar burgerlijk recht of publiekrechtelijke aanstelling arbeid laat verrichten;
- b. werknemer: de ander, bedoeld in onderdeel a;

¹⁶⁵ **Wet arbeid en zorg, artikel 3:6, eerste lid, aanhef en onderdeel a**

Voor de toepassing van deze paragraaf wordt verstaan onder werknemer: de werknemer, bedoeld in artikel 1:1, onderdeel b, met uitzondering van degene die op grond van de Eerste Afdeling, Paragraaf 2, van de Ziektewet geen werknemer in de zin van die wet is;

¹⁶⁶ In this specific case, the first definition is not completely out of effect, as part of it is re-used through a reference in the second definition.

¹⁶⁷ **Mijnbouwwet, artikel 21, vijfde lid, aanhef en onderdeel a**

De vergunning vervalt van rechtswege: a. als de houder een natuurlijke persoon is, met ingang van de dag na die waarop die persoon is overleden;

Often, no demonstrative is used, and the difference is shown by the fact that the article *the* is used instead of *a*, indicating that the text deals with a class that has been introduced earlier. For example:

Income Tax Act, article 4.50, sub 1 and 2¹⁶⁸

1. The inspector sets the amount of a loss from substantial interest by means of a decision, which may be objected to.
2. The inspector makes the decision meant at the same time that he sets the assessment for the year in which the loss arose.

In these sentences, *the decision meant* refers back to *a decision* (in this case, the reference is made more explicit by means of the addition of the word *meant*), and *the loss* refers back to *a loss*.

So, if an indefinite article (i.e. *a* or *an*) is used, the text refers to the class referred to by the noun, whereas if a definite article (i.e. *the*) is used, the text may refer to a specific subclass. This specific subclass has been introduced before, and this occurrence of that concept should be linked to the earlier occurrence.

However, it is possible that the specific subclass has not explicitly been introduced before, but has been introduced through some relation, such as *the owner of a steam apparatus*. Here, *the owner* does not refer back to *an owner* that has been introduced before, but to *a steam apparatus*.

In many cases, though, such a relation is left implicit. For example:

Succession Act 1956, article 2, sub 3¹⁶⁹

In the case of a gift by a legal person, the place of residence of the giver is considered his home.

In this sentence, *the giver* does not refer back to a giver introduced in an earlier sentence. Instead, *a gift* implies that there is a giver (and a recipient). Thus, by introducing *a gift*, the corresponding giver and recipient are also introduced. This type of integration can only be made if this implicit knowledge is added, and cannot be made automatically based on the legal text alone.

A related situation arises if the relation is made explicit in the definition. For example:

¹⁶⁸ **Wet Inkomstenbelasting 2001, artikel 4.50, eerste en tweede lid**

1. De inspecteur stelt het bedrag van een verlies uit aanmerkelijk belang vast bij voor bezwaar vatbare beschikking.
2. De inspecteur geeft de bedoelde beschikking gelijktijdig met het vaststellen van de aanslag over het jaar waarin het verlies is ontstaan.

¹⁶⁹ **Successiewet 1956, artikel 2, derde lid**

In geval van schenking door een rechtspersoon wordt de plaats, waar de schenker is gevestigd, als zijn woonplaats aangemerkt.

Protection of Antarctica Act, article 1, introduction and item e¹⁷⁰

In this law and stipulations based upon it, it is understood by organiser: the natural or legal person who organises an activity from the Netherlands.

In this definition, an organiser is linked to an activity. Thus, if a sentence introduces an organiser, it also introduces an activity.

5.6.4 Example

As an example of the different links that may occur, we will consider the following obligation:

Protection of Antarctica Act, article 33, sub 1 (partial)¹⁷¹

An organiser draws up a report within six weeks after ending the activity.

Obligation	
Action	draw up
Agent	organiser
Patient	report
Condition	within six weeks after ending the activity

This obligation refers to an organiser, as it has been defined in the definition given in the previous section:

Protection of Antarctica Act, article 1, introduction and item e

In this law and stipulations based upon it, it is understood by organiser: the natural or legal person who organises an activity from the Netherlands.

The frame for this definition is:

Definition															
Definiendum	organiser														
Definiens	<table border="1"> <thead> <tr> <th colspan="2">Or</th> </tr> </thead> <tbody> <tr> <td colspan="2">natural person</td> </tr> <tr> <td colspan="2">legal person</td> </tr> </tbody> </table> <table border="1"> <thead> <tr> <th colspan="2">AgentOf</th> </tr> </thead> <tbody> <tr> <td>Action</td> <td>organise</td> </tr> <tr> <td>Patient</td> <td>activity</td> </tr> <tr> <td>Condition</td> <td>from the Netherlands</td> </tr> </tbody> </table>	Or		natural person		legal person		AgentOf		Action	organise	Patient	activity	Condition	from the Netherlands
Or															
natural person															
legal person															
AgentOf															
Action	organise														
Patient	activity														
Condition	from the Netherlands														
Scope	this law and stipulations based upon it														

As said, the *organiser* from the obligation needs to be linked to the *organiser* that is being defined. Moreover, the definition introduces a link between the *organiser* and the *activity*. The activity mentioned in the obligation is the activity linked to the organiser through this definition, and should also be linked to the definition, yielding:

¹⁷⁰ **Wet bescherming Antarctica, artikel 1, aanhef en onderdeel a**

In deze wet en de daarop berustende bepalingen wordt verstaan onder organisator: de natuurlijke of rechtspersoon die vanuit Nederland een activiteit organiseert.

¹⁷¹ **Wet bescherming Antarctica, artikel 33, eerste lid (ingekort)**

Een organisator stelt binnen zes weken na de beëindiging van de activiteit een verslag op.

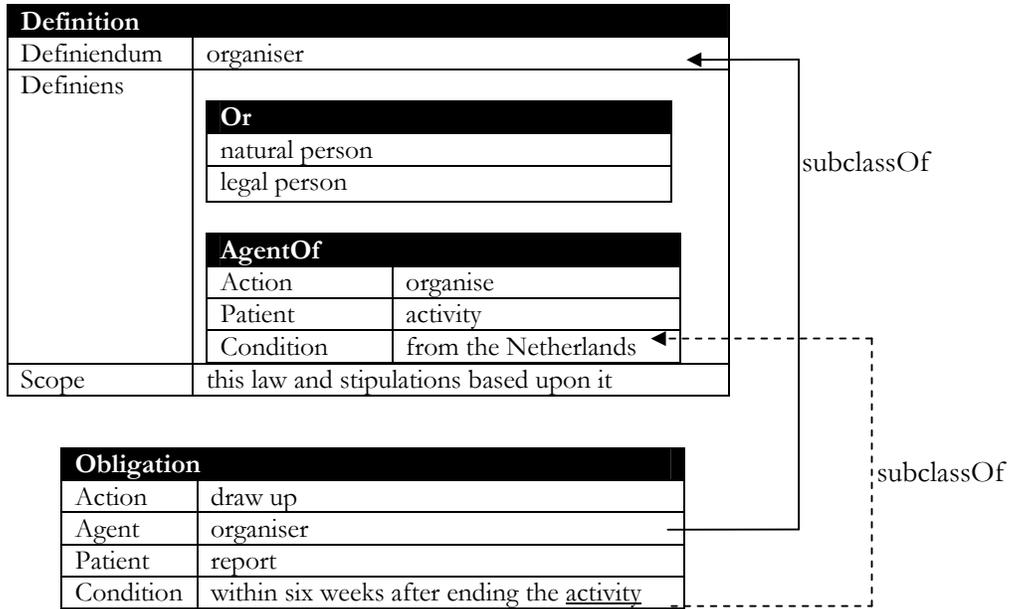


Figure 20: Linked obligation and definition frames

The next sentence refers back to the obligation given above:

Protection of Antarctica Act, article 33, sub 2 (partial)¹⁷²
 The report contains a detailed description of the performed activity.

The model of this sentence is:

Obligation	
Action	contain
Agent	report
Patient	detailed description of the performed activity

In this sentence, *the report* refers to the report mentioned in the previous obligation, and the performed activity refers back to the activity mentioned in the previous obligation. Adding this frame with those links gives:

¹⁷² **Wet bescherming Antarctica, artikel 33, tweede lid (ingekort)**
 Het verslag geeft een nauwkeurige omschrijving van de uitgevoerde activiteit.

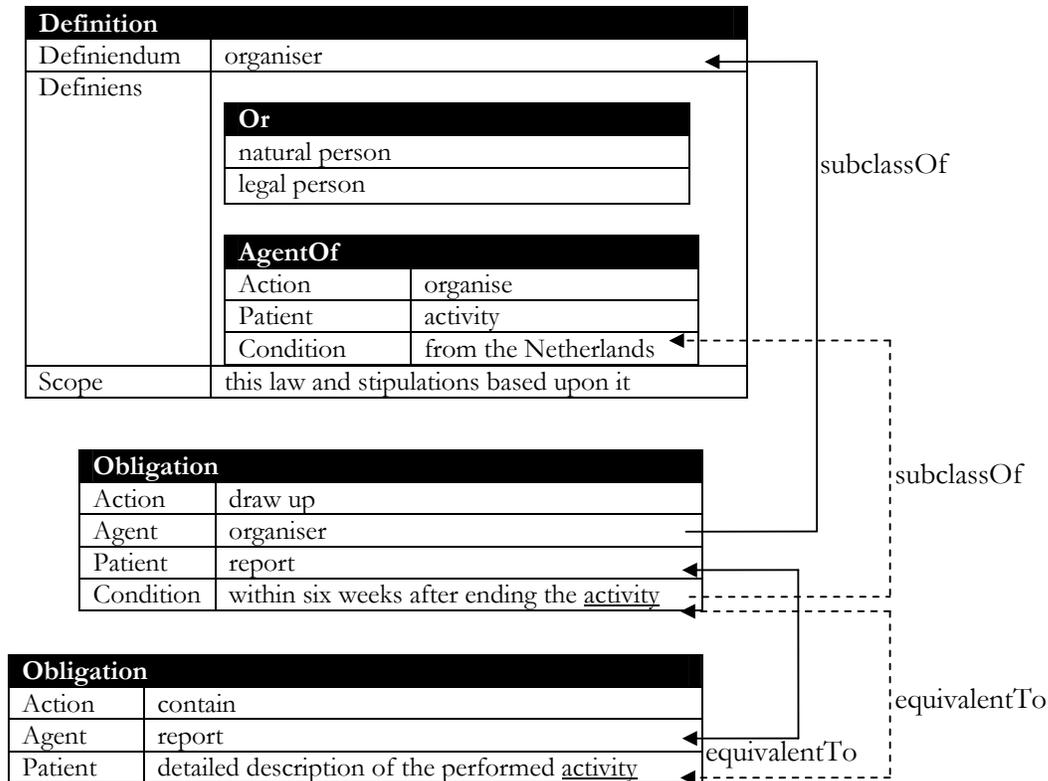


Figure 21: Linked obligation and definition frames

5.7 Conclusion

There is a lot of work still to be done. A large part of that lies in the actual implementation of the separate sentence models, but within the models for norms, there is also room for improvement. Currently, within the models, all attributes are generated as Boolean attributes. For example, an adjective like “blue” results in a condition testing whether something is blue or not blue. Human experts might create an attribute “colour” and test whether the colour is blue. As our proposed method looks only at the words and their role in the sentence, and not at their meaning, it is not able to distinguish such things. It may be possible to create lists of words for which such an intermediate concept (such as “colour”) is appropriate.

Most existing formal languages do not have support for all the different constructs that occur in a legal text. Most notably, there is no isomorphic translation for application provisions, deeming provisions and scope statements for definitions. Though such statements can be represented in most languages, there is usually some loss of information. For example, a deeming provision will be represented in the same way as a definition. When evaluating a case based upon such a representation, the outcome will be correct, but the knowledge that a deeming provision was involved can be useful in order to explain the reasoning for that outcome to the user. As such, it is desirable to have a formal, executable modelling language capable of supporting such constructs. Within the POWER project, it has been attempted to do so by adding custom constructs to UML, but these models were not executable.