This paper is the author's draft and has now been officially published as:

Beuls, K., Van Eecke, P. & Cangalovic, V. (2021). A computational construction grammar approach to semantic frame extraction. Linguistics Vanguard, 7(1). https://doi.org/10.1515/lingvan-2018-0015.

A computational construction grammar approach to semantic frame extraction

Katrien Beuls, Paul Van Eecke and Vanja Sophie Cangalovic

This paper introduces a novel methodology for extracting semantic frames from text corpora. Building on recent advances in computational construction grammar, the method captures expert knowledge of how semantic frames can be expressed in the form of conventionalised form-meaning pairings, called constructions. By combining these constructions in a semantic parsing process, the frame-semantic structure of a sentence is retrieved through the intermediary of its morpho-syntactic structure. The main advantage of this approach is that state-of-the-art results are achieved, without the need for annotated training data. We demonstrate the method in a case study where causation frames are extracted from English newspaper articles, and compare it to a commonly used approach based on Conditional Random Fields (CRFs). The computational construction grammar approach yields a word-level F_1 score of 78.5%, outperforming the CRF approach by 4.5 percentage points.

1 Introduction

Semantic frames, as originally introduced by Charles Fillmore (see e.g. Fillmore, 1982), capture a coherent part of the meaning of a sentence in a structured way. A semantic frame is defined by its *name*, which denotes the prototypical event that it represents, and by a number of *frame elements* that describe the prototypical participants in this event. For instance, the TEXT_CREATION frame, as documented in the FrameNet project (Baker et al., 1998), denotes a situation in which a new text is created, involving an AUTHOR and a TEXT as prototypical participants. In a particular linguistic utterance, this frame can be evoked by a number of different *lexical units*, including 'write', 'type' and 'draft'. Consider for example the utterance "my colleague drafted a ground-breaking paper". In this utterance, the TEXT_CREATION frame is triggered by the verb "drafted", which is therefore called the *frame-evoking element*. The AUTHOR slot in the frame is filled by "my colleague", while the TEXT slot is filled by "a ground-breaking paper". Figure 1 shows a graphical representation of this semantic frame instance, including the name of the frame, the frame-evoking element, and the two frame elements.

Semantic frame extraction (SFE) is a natural language understanding (NLU) task that consists in identifying all instances of selected semantic frames in a given text. SFE is of great

text_creation	
frame-evoking element	**drafted**
author	**my colleague**
text	**a groundbreaking paper**

Figure 1: An instance of the TEXT_CREATION frame evoked by the utterance "my colleague drafted a ground-breaking paper". The frame-evoking element is "drafted", and the AUTHOR and TEXT slots are filled by "my colleague" and "a ground-breaking paper" respectively.

relevance for many other NLU tasks, including relation extraction (Harabagiu et al., 2005), question answering (Shen and Lapata, 2007; He et al., 2015) and paraphrasing (Ellsworth and Janin, 2007). Despite the high impact that reliable SFE systems could have in the NLU community, semantic frame extraction remains an unsolved problem. This is partly due to the fact that only few reasonably large datasets with semantic frame annotations are available. The absence of such resources can be explained by the difficult nature of annotating corpora with semantic frames, therefore a time-consuming and costly endeavour, as reported by Marzinotto et al. (2018).

In this paper, we propose a novel methodology for semantic frame extraction, which does not require any annotated training data. Our approach exploits the natural relationship between frame semantics and construction grammar (Fillmore, 1988) by capturing expert knowledge of how semantic frames can be expressed in the form of form-meaning mappings, called constructions. The constructions are formalised and implemented in Fluid Construction Grammar (FCG) (Steels, 2011), a computational platform that allows using these constructions for semantically parsing linguistic utterances, in this case retrieving relevant parts of their frame-semantic structure. We evaluate the method in a case study where instances of the FrameNet CAUSATION frame are extracted from a corpus of English newspaper articles, and compare it to a commonly used method that makes use of Conditional Random Fields (CRFs).

The remainder of this paper is structured as follows. Section 2 provides a precise definition of the problem, discusses the data collection and annotation processes, presents the computational construction grammar and CRF approaches, and defines the evaluation criteria. Section 3 presents the results yielded by both approaches. The results are then discussed in Section 4 and situated with respect to earlier work in Section 5.

2 Materials and Methods

2.1 Problem Definition

In the previous section, we have introduced semantic frame extraction as a task that consists in 'identifying all instances of selected semantic frames in a given text'. This contrasts with *frame-semantic parsing*, which aims to provide a complete frame-semantic analysis of a given utterance[1]. It also differs from *semantic role labelling*, which, like semantic frame extraction, aims to extract semantic frame-like structures from utterances, but focusses on more abstract roles that describe an utterance's argument structure, such as AGENT, PATIENT and EXPERIENCER.

Following Marzinotto et al. (2018), we define semantic frame extraction as a sequence labelling problem, where zero or more labels are associated with each token in a given text. Such a label indicates that the token takes part in an instance of one of the selected semantic frames and contains information on: (i) which semantic frame the token is associated to, (ii) which semantic frame instance the label is associated to, (iii) whether the token is part of a frame-evoking element or part of a frame element, and (iv) to which frame element or lexical unit it belongs. Note that a single token can be associated to multiple frame instances and can thus receive multiple labels.

Table 1 shows an example of this sequence labelling task for a single sentence. Each token in the sentence appears on a separate row. The first column contains the index of the token in the text and the second column contains the token itself. The third column contains the frame annotation labels associated to this token. If the token is not part of one of the selected semantic frames, the default label 'O' is assigned, as is for example the case for the word 'journalists' in the example. If a token is part of a semantic frame, its label consists of 4 segments, separated by a

[1]The same distinction is made by Marzinotto et al. (2018), but referred to as *full text parsing* vs. *partial parsing*.

Table 1: Semantic frame extraction is defined as a sequence labelling task, in which the frame annotation labels need to be predicted.

Index	Token	Frame Annotation
798	Back	O
799	in	O
800	1984	O
801	,	O
802	journalists	O
803	reported	O
804	from	O
805	Ethiopia	O
806	about	O
807	a	Causation:FE:effect:812
808	famine	Causation:FE:effect:812
809	of	Causation:FE:effect:812
810	biblical	Causation:FE:effect:812
811	proportions	Causation:FE:effect:812
812	caused	Causation:LU:cause:812
813	by	O
814	widespread	Causation:FE:cause:812
815	drought	Causation:FE:cause:812
816	.	O

colon. The first segment provides the name of the semantic frame that the token belongs to (e.g. 'causation'). The second segment indicates whether the token is part of a frame-evoking element ('LU' for lexical unit) or of a frame element ('FE'). If it is part of a frame-evoking element, as is the case for the word 'caused' in the example, the third segment of the label reveals the lexical unit that triggered the frame ('cause'). If the token is part of a frame element, as is for example the case for 'famine' and 'draught', the third part of the label indicates the specific frame element it belongs to, EFFECT and CAUSE respectively in this case. Finally, the fourth segment of the label refers to the index of the token that triggered the frame instance, in this case 'caused' on line 812. The index allows identifying to which frame instance a frame annotation label belongs, which is especially important in sentences where multiple frame instances occur next to each other.

2.2 Data Collection

The semantic frame extraction method that we present in this paper, was initially developed as part of an application that analyses opinions about causal relations expressed in online media, especially focussing on the climate change debate. The frame of interest here is the CAUSATION frame, with as frame elements CAUSE and EFFECT. The texts from which the frames need to be extracted are online newspaper articles and their comments. We use this case study in the coming sections for demonstrating and evaluating the frame extraction method.

The first step in the data collection process consisted in the compilation of a corpus of online newspaper articles, more in particular articles from the *The Guardian* that are tagged with the topic 'Climate Change'. From this corpus, a subcorpus of 'causal' sentences was selected, based on the following two criteria: (i) each sentence should contain at least one of the following lexical units, listed in FrameNet as frame evoking elements of the CAUSATION frame: CAUSE.V, DUE TO.PREP, BECAUSE.C, BECAUSE OF.PREP, GIVE RISE TO.V, LEAD TO.V or RESULT IN.V, and (ii) each of these lexical units should appear at least 40 times across all sentences in the subcorpus. The sentences were otherwise randomly selected. The resulting subcorpus consisted of 345 sentences with a total of 11443 words, amounting to an average of 33 words per sentence.

Table 2: Corpus statistics and composition of development and test set.

Set	# Sentences	# Words	# Frames
Dev. Set	199	6555	241
Test Set	146	4888	171
All	**345**	**11443**	**412**

Table 3: Overview of frame instances evoked by different lexical units in the corpus.

Lexical Unit	Dev. Set	Test Set	Total
CAUSE.V	55	31	86
DUE-TO.PREP	33	23	56
BECAUSE.C	48	21	69
BECAUSE-OF.PREP	20	21	41
GIVE-RISE-TO.V	20	30	50
LEAD-TO.V	35	23	58
RESULT-IN.V	30	22	52
All	**241**	**171**	**412**

These sentences contain in total 412 lexical units that evoke an instance of the CAUSATION frame.

These 345 sentences were then transformed into an automatically annotated CONLL-like format using the Spacy NLP toolbox[2]. In this format, each token appears on a separate line, together with its lemma, universal part-of-speech tag, dependency label, and the index of the token that serves as its head in the sentence's syntactic dependency tree. Each token was also manually annotated with zero or more frame labels, as explained in Section 2.1 above and illustrated in Table 1. In total, 412 instances of the CAUSATION frame were annotated, using 6317 labels, 735 of these associated to frame evoking elements, 2183 to causes and 3399 to effects.

The annotated corpus was then subdivided into a development set and a test set, respectively covering 60% and 40% of the frame instances. The subdivision was done randomly, with the only constraint that each lexical unit needed to appear at least 20 times in each set. As shown in Table 2, 199 sentences with 241 frame instances were assigned to the development set, while 146 sentences with 171 frame instances were assigned to the test set. Table 3 presents the distribution of the different lexical units in the corpus, showing that each lexical unit appears at least 20 times in both the development and the test set.

Table 4 shows a sample of the annotated corpus, illustrating the data format for the sentence "Loss of Arctic sea ice results in enhanced warming of the Arctic Ocean due to a strong positive feedback". The first column indicates the index of the token in the corpus ('Index'). Column two to six are automatic annotations of the index of the token in the sentence ('Id'), the word form of the token ('Token'), its lemma ('Lemma'), its universal part-of-speech tag ('Upos'), the index of its head in the sentence's dependency tree ('H'), and the token's dependency label ('Deprel'). The last column shows the manually annotated frame labels. The example sentence contains two instances of the CAUSATION frame, one evoked by 'results in' on line 5651 and one evoked by 'due to' on line 5659. The indices in the annotated frame labels are important for indicating with which frame instance each label is associated. For instance, the token "Loss" on line 5646 is part of the cause of the frame instance evoked on line 5651 comprising "[$_{\text{CAUSE}}$ Loss of arctic sea ice] *results in* [$_{\text{EFFECT}}$ enhanced warming of the Arctic Ocean']', but at the same time part of the effect of the frame instance evoked on line 5659 comprising "[$_{\text{EFFECT}}$ Loss of Arctic sea ice results in enhanced warming of the Arctic Ocean] *due to* [$_{\text{CAUSE}}$ a strong positive feedback]".

The annotated corpus is available on request.

[2]Spacy v2.0.18 – `https://spacy.io`

Index	Id	Form	Lemma	Upos	H	Deprel	Frame labels	
5646	1	Loss	loss	noun	_	ROOT	C:FE:cause:5651	C:FE:effect:5659
5647	2	of	of	adp	1	prep	C:FE:cause5651	C:FE:effect:5659
5648	3	Arctic	arctic	propn	4	compound	C:FE:cause:5651	C:FE:effect:5659
5649	4	sea	sea	noun	6	compound	C:FE:cause:5651	C:FE:effect:5659
5650	5	ice	ice	noun	6	compound	C:FE:cause:5651	C:FE:effect:5659
5651	6	results	result	noun	2	pobj	C:LU:result-in:5651	C:FE:effect:5659
5652	7	in	in	adp	6	prep	C:LU:result-in:5651	C:FE:effect:5659
5653	8	enhanced	enhanced	adj	9	amod	C:FE:effect:5651	C:FE:effect:5659
5654	9	warming	warming	noun	7	pobj	C:FE:effect:5651	C:FE:effect:5659
5655	10	of	of	adp	9	prep	C:FE:effect:5651	C:FE:effect:5659
5656	11	the	the	det	13	det	C:FE:effect:5651	C:FE:effect:5659
5657	12	Arctic	arctic	propn	13	compound	C:FE:effect:5651	C:FE:effect:5659
5658	13	Ocean	ocean	propn	10	pobj	C:FE:effect:5651	C:FE:effect:5659
5659	14	due	due	adp	1	prep		C:LU:due-to:5659
5660	15	to	to	adp	14	pcomp		C:LU:due-to:5659
5661	16	a	a	det	19	det		C:FE:cause:5659
5662	17	strong	strong	adj	19	amod		C:FE:cause:5659
5663	18	positive	positive	adj	19	amod		C:FE:cause:5659
5664	19	feedback	feedback	noun	14	pobj		C:FE:cause:5659
5665	20	.	.	punct	1	punct	O	

Table 4: Sample of the corpus with annotations for the CAUSATION frame. Two frames are evoked in this sentence: one by the lexical unit RESULT-IN.V at index 5651 and one by DUE-TO.PREP at index 5659. In the frame labels, 'Causation:' has be shortened to 'C:' for space reasons. The frame labels are here indented for clarity, but the order in which they appear for a single token is not significant. It is the indexes that indicate to which frame instance the label is associated.

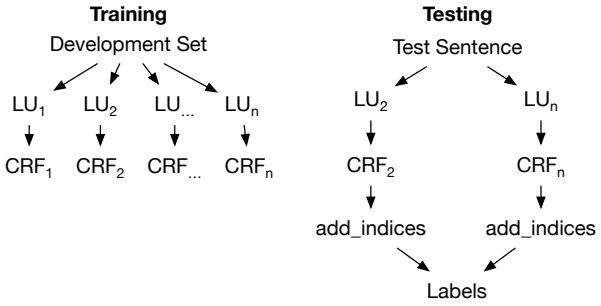

Figure 2: Artchitecture of the CRF baseline system, based on Marzinotto et al. (2018). At training time, a separate CRF is trained for each lexical unit (LU) on sentences that contain that LU. At test time, a sentence is input to the CRFs trained on the LUs that it contains, the indices are added, and the predictions of the different CRFs are merged.

2.3 Conditional Random Field Baseline

Before presenting our novel approach to semantic frame extraction, we first build a baseline system, which will later allow us to better interpret our systems evaluation results. We use an approach based on conditional random fields (CRFs), which have often been used in the NLP literature for solving similar sequence labeling tasks (McCallum and Li, 2003; Cohn and Blunsom, 2005; Marzinotto et al., 2018), and perform relatively well on small datasets. Concretely, our baseline system implements the CRF-based semantic frame extraction method presented by Marzinotto et al. (2018).

Following this approach, a separate CRF is trained for each lexical unit. In our case, this amounts to six CRFs in total (see Table 3 - BECAUSE.C and BECAUSE-OF.PREP are handled by the same CRF because of their lexical similarity). Before training, a subset of the development set is selected for each lexical unit, consisting of only those sentences in which the lexical unit appears. This is done based on the lemma of the word that is most strongly associated with the LU: 'cause', 'due', 'because', 'rise', 'lead', and 'result' respectively. Then, each CRF is trained on one of these subsets, with as features the lemma of each token, its universal part-of-speech tag, and an encoding of the shortest path in the dependency tree between the token and the first word of the first occurrence of the LU in the sentence. For instance, the features of the word 'positive' for training the CRF dealing with the LU DUE-TO in the example shown in Table 4 would be ['positive', 'adj', '+amod+pobj'], with the + signs indicating the upwards direction of the described links in the dependency tree. The labels for training the CRFs are the annotated frame labels that are associated with all frame instances triggered by the LU, but without their indices (e.g. C:FE:cause).

At test time, a preprocessor first identifies the possible LUs in a sentence based on their lemma, using the same methodology that was used for creating the training subsets for each LU. Then, the sentence is input to the CRFs that were trained for these LUs, and the predictions are collected. The indices identifying the frame instances are then added to the predicted labels in a very naive way, namely by referring to the index of the first word of the first occurrence of the LU in the sentence. Finally, the predictions of the individual CRFs are merged, with the effect that each token can become associated with multiple frame labels. An overview of the architecture of the CRF baseline system is shown in Figure 2.

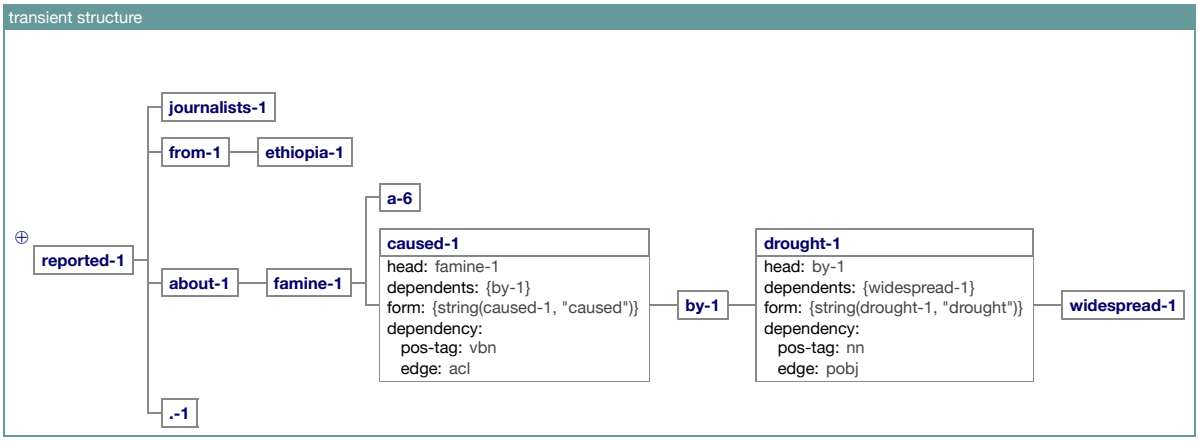

Figure 3: Dependency structure for the sentence "Journalists reported from Ethiopia about a famine caused by widespread drought". This structure is forms the input for the constructional processing step. Two nodes are expanded, showing more details of the information contained in the structure.

2.4 Computational Construction Grammar Approach

Computational construction grammar (CCxG) is a branch of linguistics that operationalises insights and analyses from construction grammar into concrete processing models (Van Eecke and Beuls, 2018). As such, CCxG aims to capture linguistic knowledge about the morphology, syntax and semantics of a language in the form of form-meaning pairings, called constructions. These constructions can range from very concrete, for example connecting a word form to its lexical meaning, to completely abstract, for example in the case of an argument structure construction that combines a subject noun phrase with an intransitive ergative verb, and specifies that the subject is the undergoer of the event evoked by the verb. In this paper, we use Fluid Construction Grammar (FCG) (Steels, 2011) for formally representing and processing constructions, with the goal of extracting semantic frames from text corpora.

Concretely, our novel CCxG-based semantic frame extraction method proceeds in three steps. First, a syntactic dependency structure is built based on the corpus annotation. Then, targeted constructions that encode expert linguistic knowledge extend the dependency structure with information about the presence of semantic frames and their frame elements. Finally, this information is extracted from the resulting structure and transformed into frame labels.

We will now explain the approach in more detail using the example sentence "Journalists reported from Ethiopia about a famine caused by widespread drought". Figure 3 shows the dependency structure of the sentence as annotated in the corpus, in a feature structure format that can easily be used by the FCG engine. Every node contains 4 features. The `head` and `dependents` features encode the structure of the tree, referring to the head and dependents of a node by their unique identifier. The `form` feature contains the word form of the token that a node represents. Finally, the `dependency` feature holds the part-of-speech tag of the node, as well as the edge label of the node to its head.

Next, the dependency structure is extended by 3 constructions: the CAUSED-MORPH-CXN, the CAUSE-VERB-LEX-CXN and the X-CAUSED-BY-Y-CXN. The CAUSED-MORPH-CXN searches in the dependency structure for a node of which the `form` feature contains the string "caused". If such a node is found, the construction will add two new features to this node: a `lex-class` feature with the value `verb` and a `lex-id` feature with the value `cause`. This construction identifies thus that the token is a morphological form of the verb 'to cause'. Now that these features are part of the structure, other constructions can build further on them. Figure 4 shows the FCG

representation of the CAUSED-MORPH-CXN. The preconditions of the construction are written on the right side of the arrow, while features that are added are written on the left side[3]. Symbols preceded by a question mark are logical variables, which can match any symbol in the structure.

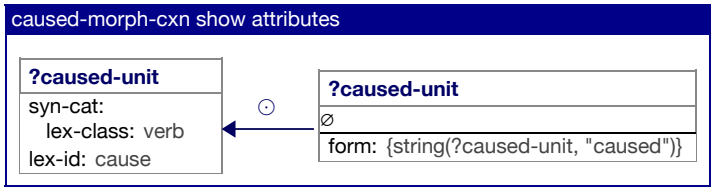

Figure 4: The CAUSED-MORPH-CXN adds information about the lemma and lexical class of the word form "caused" to the dependency structure.

The CAUSE-VERB-LEX-CXN is a lexical construction that looks for word forms of the verb 'to cause' and identifies them as frame-evoking elements of the CAUSATION frame. In this case, the construction matches on the features that were added by the CAUSED-MORPH-CXN. The CAUSE-VERB-LEX-CXN then adds a `meaning` feature that holds three predicates. The first predicate indicates the name of the frame (CAUSATION), the lexical unit that triggered it ('cause'), and a unique identifier for the frame instance (`?frame`). The second and third predicates represent the slots of the frame. Their first argument specifies the participant role that they represent (CAUSE and EFFECT respectively). Their second argument links the predicates to the frame instance they belong to (`?frame`). Their third argument is for now a free variable, but will eventually be linked to the respective referents of the participant roles in the dependency structure. Finally, the `sem-valence` feature abstracts away from the specific semantic roles of `cause` and `effect` to the more general roles of `actor` and `theme`. These more general roles allow for a higher degree of generalisation in the grammar, avoiding the need to write specific constructions for every frame.

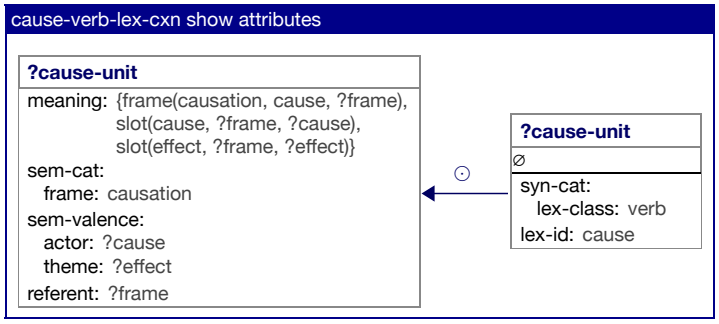

Figure 5: The CAUSE-VERB-LEX-CXN identifies f CAUSATION frames evoked by the lexical unit CAUSE.V.

The X-CAUSED-BY-Y-CXN matches on four nodes in the structure, referred to as `?caused-unit`, `?by-unit`, `?cause-unit` and `?effect-unit`. The `?caused-unit` matches on the node that has previously been identified by the CAUSE-VERB-LEX-CXN as a frame-evoking element for the CAUSATION frame. Moreover, it needs to have the part-of-speech tag `vbn` and the edge label `acl` under its `dependency` feature. The `?by-unit` needs to be a daughter node of the `?caused-unit`, that contains the string "by". In turn, the `?cause-unit` needs to be a daughter node of the `?by-unit`, and have the edge label `pobj`. Finally, the `?effect-unit` is the node that is the head of the `?caused-unit`. If these preconditions are fulfilled, the construction adds a `referent`

[3]Note that the nodes appearing on the right side of the arrow (preconditions) are divided into two by a horizontal line. Only the part below the line is used. The part above the line is reserved for preconditions for producing sentences, which is not part of the task here.

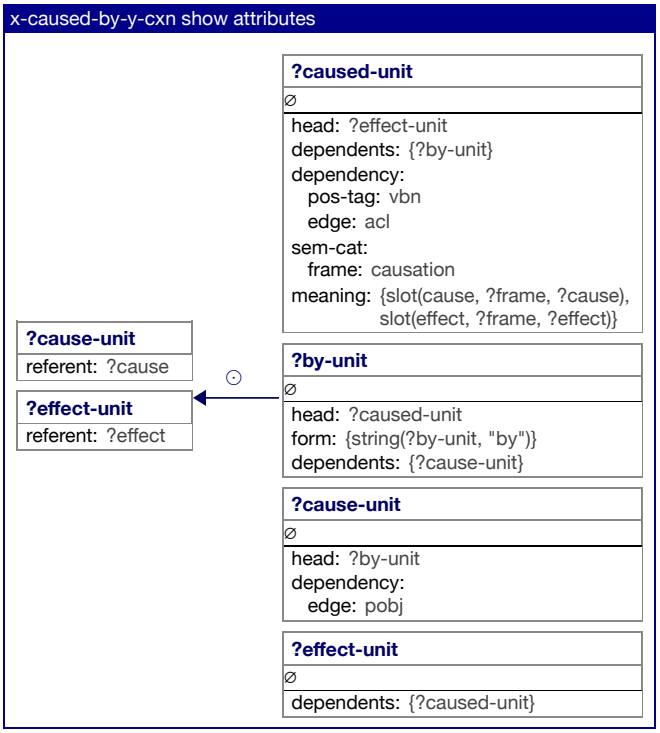

Figure 6: The X-CAUSED-BY-Y-CXN fills in the participant roles of the frame, based on their syntactic properties.

feature to the `?cause-unit` and `?effect-unit` nodes. The values of these features are the same variables as those that appear in the frame slot predicates of the `?caused-unit`, indicating that these nodes serve as the referent of the CAUSE and EFFECT slots of the frame.

Figure 7 shows the structure of the sentence after it has been extended by the three constructions discussed above. The variable links indicate that the `caused-1` node evokes an instance of the CAUSATION frame, and that the `drought-1` and `famine-1` nodes are the head of the CAUSE and EFFECT slots respectively. Now, these relations need to be translated into frame labels. For the heads of both slots (`drought-1` and `famine-1`), the system looks up all daughter nodes that are not part of a subtree of which the top node does not contain a `referent` feature. In this case, these are the `widespread-1` node for the cause and the `a-1` node for the effect. In the corpus, a frame label is added to all tokens represented by both the head node and the selected daughter nodes. The first element of the label is the name of the evoked frame, as specified by the frame predicate in the frame-evoking node. The second element is 'LU' for the tokens represented by the frame-evoking node and its selected daughter nodes, and 'FE' for the tokens belonging to the participant roles. The third element is the second argument of the frame predicate for LU tokens, and the first argument of the slot predicate for the FE tokens. Finally, the index is the sum of the position of the first token of the LU in the sentence with the set-off of the start of the sentence in the corpus. Note that multiple frame instances in the same sentence, even if they are nested as in the case of the example in Table 4, form no obstacle for this approach.

The grammar used for this case study was developed based on the development part of the corpus. In total, it consists of 58 constructions. 17 morphological constructions identify instances of lexical units that exhibit morphological variation. 11 lexical constructions identify frame instances. Finally, 30 grammatical constructions fill in the slots of these frame instances. Some of these are specific to the CAUSATION frame, for example the X-CAUSED-BY-Y-CXN shown in Figure 6, while others are more generally applicable, such as the ACTIVE-TRANSITIVE-CXN.

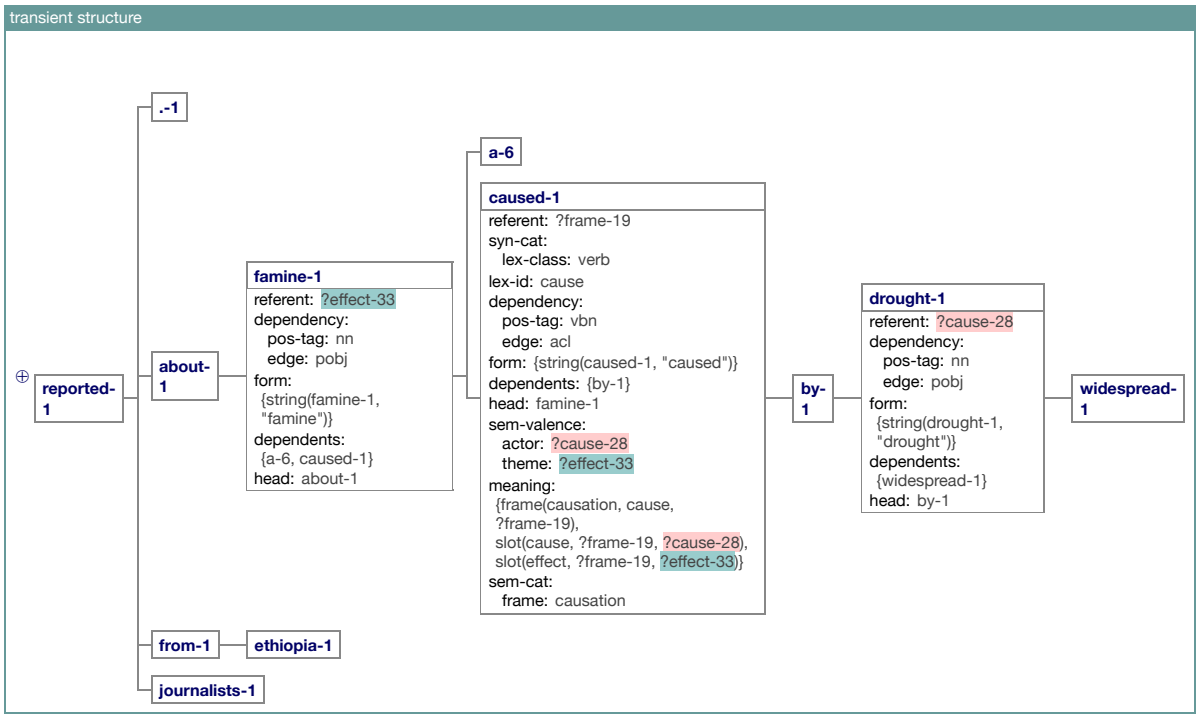

Figure 7: The dependency structure has been extended by the CAUSED-MORPH-CXN, CAUSE-VERB-LEX-CXN and the X-CAUSED-BY-Y-CXN. The colouring highlights that the `caused-1` node evokes an instance of the CAUSATION frame, and that the `drought-1` and `famine-1` nodes are the head of the CAUSE and EFFECT slots respectively.

2.5 Evaluation Criteria

Both the CRF baseline system and the computational construction grammar approach are evaluated on the test set described in Section 2.2, where their predictions are compared against the ground-truth annotations. As measures for evaluation, we use *precision*, calculated by dividing the number of correctly predicted frame labels by the total number of predicted frame labels, *recall*, calculated by dividing the number of correctly predicted frame labels by the total number of annotated frame labels, and F_1 *score* as a measure of accuracy, calculated as the harmonic average of precision and recall, as shown in Equation (1).

$$F_1 = 2 \cdot \frac{precision \cdot recall}{precision + recall} \tag{1}$$

Precision, recall and F_1 score are calculated on multiple levels of granularity: for all frame labels, for those relating to frame evoking elements, for those relating to frame elements, for those relating to causes and for those relating to effects.

3 Results

An overview of the results of the evaluation of the CRF baseline system and the computational construction grammar approach on the test portion of the newpaper corpus is presented in Table 5. The first column indicates the unit of evaluation: either the labels relating to frame evoking elements (LU), frame elements (FE), causes (Cause) or effects (Effect), or all labels taken together (Overall). The second column shows the method that was used, either conditional random fields (CRF) or computational construction grammar (CCxG). The last three columns

Unit	Method	Precision	Recall	F_1
LU	CRF	98.08	95.62	96.84
	CCxG	95.70	90.31	92.93
FE	CRFs	75.71	66.53	70.82
	CCxG	85.27	69.24	76.42
Cause	CRFs	83.27	73.21	77.92
	CCxG	85.75	75.84	80.49
Effect	CRFs	71.19	62.18	66.38
	CCxG	85.13	64.68	73.51
Overall	CRF	78.60	69.95	74.02
	CCxG	**86.67**	**71.72**	**78.49**

Table 5: Evaluation results of the computational construction grammar (CCxG) and conditional random field (CRF) methods on the Guardian newspaper corpus. LU: frame evoking elements, FE: frame elements.

reveal the results for precision, recall and F_1 score respectively.

Overall, the CCxG approach yields a precision of 86.57%, a recall of 71.72% and an F_1 score of 78.49%, outperforming the CRF approach on all three metrics, with 8.07, 1.77 and 4.47 percentage points respectively. Also on the level of the frame elements, the CCxG approach yields a better precision, recall and F_1 score than the CRF baseline (85.27%, 69.24%, 76.42% against 75.71%, 66.53%, 70.82%), a result that is also true for both causes (85.75%, 75.84%, 80.49% against 83.27%, 73.21%, 77.92%), and effects (85.13%, 64.68%, $73,51\%$ against 71.19%, 62.18%, $66,38\%$). On the level of the lexical units however, the CRF baseline performs better than the CCxG approach (98.08%, 95.62%, $96,84\%$ against 95.70%, 90.31%, $92,38\%$).

4 Discussion

The results have shown that the CCxG approach performs considerably better than the CRF baseline on almost all metrics and units of evaluation, except when it comes to the identification of frame-evoking elements. The difference is especially pronounced for the precision scores of effect identification, where the CCxG approach clearly outperforms the baseline by almost 14 percentage points. While causes are often expressed as the subject of causal verbs, and are thus relatively easy to identify, effects have more diverse syntactic realisations in the corpus. This is handled by the CCxG approach through the implementation of different constructions for each of these realisations, while the CRF approach might be more susceptible to sparseness here. The observation that the CRF approach performs better than the CCxG approach on the relatively simple task of frame identification, can be explained by the fact that our system does not return frame labels for frame instances where no slots could be filled. This is a design choice that is motivated by the current applications of the system, for which frame identification in its own is not a goal.

The main advantage of the CCxG approach is that it does not rely on annotated training data. Only a small held-out development set is needed to serve as an inspiration to the grammar engineer, who is able to include expert linguistic knowledge of how semantic frames can be expressed into the grammar. The CRF approach on the other hand, relies entirely on training data annotated with semantic frames, which is only scarcely available. It is likely that the results of the CRF approach will still improve when more data is used for training, although this remains a hypothesis that cannot be concluded from our results.

When it comes to the effort involved in the development of a semantic frame extraction tool, the CRF approach is less time-consuming and might be preferred when enough annotated training data is available. Implementing a CCxG grammar requires more time and a basic level

of linguistic knowledge of the target language, but is surely much more feasible than annotating a training corpus by hand, and the effort is rewarded by a good accuracy score on the task.

5 Related Work

The fields of frame-semantic parsing and semantic frame extraction are dominated by approaches that make use of (semi-)supervised machine learning techniques, including discriminative (Gildea and Jurafsky, 2002; Das et al., 2014) and generative (Thompson et al., 2003) statistical models, maximum entropy models (Fleischman et al., 2003), support vector machines (SVMs) (Giuglea and Moschitti, 2006; Johansson and Nugues, 2007), conditional random fields (Marzinotto et al., 2018) and bi-directional LSTMs (Ringgaard et al., 2017; Marzinotto et al., 2018). Given that these models need to be trained on annotated corpora, which are only scarcely available, they are almost always applied to English and typically trained on FrameNet's example sentences. An exception to this is a recent contribution by Marzinotto et al. (2018), who created their own training data for French, but only included a limited number of frames.

When it comes to grammar-based approaches to frame-semantic parsing, prior work is much more sparse. Shi and Mihalcea (2004) present a broad-coverage, rule-based system that combines knowledge from FrameNet and WordNet with handwritten semantic mapping rules. Their system is evaluated on a subset of FrameNet's example sentences, but the results are hard to interpret as no detailed evaluation is reported. In the computational construction grammar literature, two different approaches have been proposed. Micelli et al. (2009) present an FCG grammar that incorporates the FrameNet lexicon and a few handwritten grammatical constructions. Dodge et al. (2017) employ Embodied Construction Grammar (ECG) to process a grammar that consists of constructions that were automatically generated based on FrameNet valence patterns. However, both papers only describe initial proofs of concept, which are not intended for broad-coverage parsing and have therefore not been evaluated.

Finally, Dunietz et al. (2017) present two promising approaches that combine automatically induced pattern-matching rules with statistical classifiers for extracting causal relations. While these approaches are inspired by ideas from construction grammar, they rely on supervised machine learning techniques and consequently on considerable amounts of annotated training data.

6 Conclusion

In this paper, we have introduced a novel methodology for extracting semantic frames from text corpora. The method combines broad-coverage dependency parsing with lexical and grammatical constructions that capture expert knowledge of how specific semantic frames can be expressed. The main advantage of the approach is that state-of-the-art results are achieved without the need for annotated training data.

We have demonstrated the methodology in a case study where causation frames were extracted from English newspaper sentences. The computational construction grammar approach outperformed a conditional random fields baseline by 4.5 percentage points in F_1 score.

In sum, the proposed computational construction grammar-based methodology forms an excellent alternative to machine learning techniques, especially given the fact that frame-annotated training data is expensive to create and only scarcely available.

Acknowledgements

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 732942, from the Flemish Government under the

'Onderzoeksprogramma Artificiële Intelligentie (AI) Vlaanderen' programme and from a post-doctoral fellowship of the Research Foundation Flanders (FWO) awarded to PVE (grant No 75929). We would like to thank Luc Steels for his valuable feedback on this work and Remi van Trijp as area editor for *Linguistics Vanguard*.

References

Baker, C., Fillmore, C., and Lowe, J. (1998). The Berkeley framenet project. In *Proceedings of the 17th international conference on Computational linguistics-Volume 1*, pages 86–90. Association for Computational Linguistics.

Cohn, T. and Blunsom, P. (2005). Semantic role labelling with tree conditional random fields. In *Proceedings of the Ninth Conference on Computational Natural Language Learning*, pages 169–172. Association for Computational Linguistics.

Das, D., Chen, D., Martins, A., Schneider, N., and Smith, N. A. (2014). Frame-semantic parsing. *Computational linguistics*, 40(1):9–56.

Dodge, E., Trott, S., Gilardi, L., and Stickles, E. (2017). Grammar scaling: Leveraging FrameNet data to increase embodied construction grammar coverage. In *2017 AAAI Spring Symposia, Stanford University, Palo Alto, California, USA, March 27-29, 2017*.

Dunietz, J., Levin, L., and Carbonell, J. (2017). Automatically tagging constructions of causation and their slot-fillers. *Transactions of the Association for Computational Linguistics*, 5:117–133.

Ellsworth, M. and Janin, A. (2007). Mutaphrase: Paraphrasing with framenet. In *Proceedings of the Workshop on Textual Entailment and Paraphrasing*, Prague. Association for Computational Linguistics, Association for Computational Linguistics.

Fillmore, C. (1982). Frame semantics. In *Linguistics in the morning calm*, pages 111–138.

Fillmore, C. (1988). The mechanisms of "construction grammar". In *Annual Meeting of the Berkeley Linguistics Society*, volume 14, pages 35–55.

Fleischman, M., Kwon, N., and Hovy, E. (2003). Maximum entropy models for framenet classification. In *Proceedings of the 2003 conference on Empirical methods in natural language processing*, pages 49–56. Association for Computational Linguistics.

Gildea, D. and Jurafsky, D. (2002). Automatic labeling of semantic roles. *Computational linguistics*, 28(3):245–288.

Giuglea, A.-M. and Moschitti, A. (2006). Semantic role labeling via framenet, verbnet and propbank. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 929–936. Association for Computational Linguistics.

Harabagiu, S., Bejan, C., and Morarescu, P. (2005). Shallow semantics for relation extraction. In *IJCAI-05, Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence, Edinburgh, Scotland, UK, July 30 - August 5, 2005*, pages 1061–1066.

He, L., Lewis, M., and Zettlemoyer, L. (2015). Question-answer driven semantic role labeling: Using natural language to annotate natural language. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 643–653.

Johansson, R. and Nugues, P. (2007). Lth: semantic structure extraction using nonprojective dependency trees. In *Proceedings of the fourth international workshop on semantic evaluations (SemEval-2007)*, pages 227–230.

Marzinotto, G., Auguste, J., Béchet, F., Damnati, G., and Nasr, A. (2018). Semantic frame parsing for information extraction : the CALOR corpus. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, pages 986–993.

McCallum, A. and Li, W. (2003). Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*, pages 188–191. Association for Computational Linguistics.

Micelli, V., van Trijp, R., and De Beule, J. (2009). Framing Fluid Construction Grammar. In Taatgen, N. and van Rijn, H., editors, *Proceedings of the 31th Annual Conference of the Cognitive Science Society*, pages 3023–3027. Cognitive Science Society.

Ringgaard, M., Gupta, R., and Pereira, F. (2017). Sling: A framework for frame semantic parsing. Technical report.

Shen, D. and Lapata, M. (2007). Using semantic roles to improve question answering. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 12–21.

Shi, L. and Mihalcea, R. (2004). An algorithm for open text semantic parsing. In *Proceedings of the 3rd Workshop on RObust Methods in Analysis of Natural Language Data*, pages 59–67. Association for Computational Linguistics.

Steels, L., editor (2011). *Design Patterns in Fluid Construction Grammar*. John Benjamins, Amsterdam.

Thompson, C., Levy, R., and Manning, C. (2003). A generative model for semantic role labeling. In *European Conference on Machine Learning*, pages 397–408. Springer.

Van Eecke, P. and Beuls, K. (2018). Exploring the creative potential of computational construction grammar. *Zeitschrift für Anglistik und Amerikanistik*, 66(3):341–355.