Catena Operations for Unified Dependency Analysis

Kiril Simov and **Petya Osenova**
Linguistic Modeling Department
Institute of Information and Communication Technologies
Bulgarian Academy of Sciences
Bulgaria
{kivs,petya}@bultreebank.org

Abstract

The notion of catena was introduced originally to represent the syntactic structure of multiword expressions with idiosyncratic semantics and non-constituent structure. Later on, several other phenomena (such as ellipsis, verbal complexes, etc.) were formalized as catenae. This naturally led to the suggestion that a catena can be considered a basic unit of syntax. In this paper we present a formalization of catenae and the main operations over them for modelling the combinatorial potential of units in dependency grammar.

1 Introduction

Catenae were introduced initially to handle linguistic expressions with non-constituent structure and idiosyncratic semantics. It was shown in a number of publications that this unit is appropriate for both - the analysis of syntactic (for example, ellipsis, idioms) and morphological phenomena (for example, compounds). One of the important questions in NLP is how to establish a connection between the lexicon and the text dimension in an operable way. At the moment most investigations focus on the representation and analysis of the text dimension.

We first employed catenae when modeling multiword expressions in Bulgarian within the relation lexicon - text. (Simov and Osenova, 2014). Encouraged by the promising results, we continued our research on how to exploit catenae as a unified strategy for dependency analysis. In the paper we use examples mostly from Bulgarian and to a lesser extend from English, but our approach is applicable to other languages, as well.

In this piece of research we pursue both issues mentioned above. On the one hand, we show in a formal way how the lexicon representation maps to its syntactic analysis. On the other hand, a unified strategy of dependency analysis is proposed via extending the catena to handle also phenomena as valency and other combinatorial dependencies. Thus, a two-fold analysis is achieved: handling the lexicon-grammar relation and arriving at a single means for analyzing related phenomena.

The paper is structured as follows: the next section outlines some previous work on catenae; section 3 focuses on the formal definition of the catena and of catena-based lexical entries; section 4 presents different lexical entries that demonstrate the expressive power of the catena formalism; section 5 concludes the paper.

2 Previous Work on Catenae

The notion of catena (chain) was introduced in (O'Grady, 1998) as a mechanism for representing the syntactic structure of idioms. He shows that for this task there is need for a definition of syntactic patterns that do not coincide with constituents. He defines the catena in the following way: *The words A, B, and C (order irrelevant) form a chain if and only if A immediately dominates B and C, or if and only if A immediately dominates B and B immediately dominates C.*

In recent years the notion of catena revived again and was applied also to dependency representations. Catenae have been used successfully for the modelling of problematic language phenomena. (Gross 2010) presents the problems in syntax and morphology that have led to the introduction of the subconstituent catena level. Constituency-based analysis faces non-constituent structures in ellipsis, idioms, verb complexes.

Apart from the linguistic modelling of language phenomena, catenae have been used in a number of NLP applications. (Maxwell et al., 2013), for example, presents an approach to Information Retrieval based on catenae. The authors consider the catena as a mechanism for semantic encoding

Proceedings of the Third International Conference on Dependency Linguistics (Depling 2015), pages 320–329,
Uppsala, Sweden, August 24–26 2015.

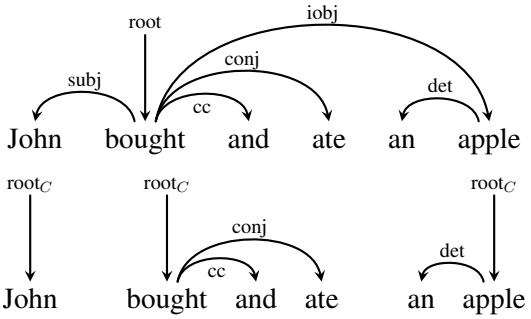


Figure 1: A complete dependency tree and some of its catenae.

which overcomes the problems of long-distance paths and elliptical sentences. The employment of catenae in NLP applications is additional motivation for us to use it in the modelling of the interface between the treebank and the lexicon.

Terminology note: an alternative term for catena is *treelet*. It has been used in the area of machine translation as a unit for translation transfer (see (Quirk et al., 2005)). Their definition is equivalent to the definition of catena. Also (Kuhlmann, 2010) uses *treelet* for a node and its children (if any). In the paper we resort to the term catena because it is closer to the spirit of the issues discussed here.

3 Formal Definition of Catena

In this section we define the formal presentation of the catena as it is used in syntax and in the lexicon. Here we follow the definition of catena provided by (O'Grady, 1998) and (Gross, 2010): a **catena** is a word or a combination of words directly connected in the dominance dimension. In reality this definition of catena for dependency trees is equivalent to a subtree definition. Fig. 1 depicts a complete dependency tree and some of its catenae. Notice that the complete tree is also a catena itself. With "root$_C$" we mark the root of the catena. It might be the same as the root of the complete tree, but also might be different as in the cases of "John" and "an apple". Following (Osborne et al., 2012) we prefer to use the notion of catena to that of dependency subtree or treelet as mentioned above. We aim to utilize the notion of catena for several purposes: representation of words and multiword expressions in the lexicon, their realization in the actual trees expressing the analysis of sentences as well as for representation of derivational structure of compounds in the lexicon.

In order to model the variety of phenomena and characteristics encoded in a dependency grammar we extend the catena with partial arc and node labels. We follow the approach taken in CoNLL shared tasks on dependency parsing representing for each node its word form, lemma, part of speech, extended part of speech, grammatical features (and later – semantics). This provides a flexible mechanism for expressing the combinatorial potential of lexical items. In the following definition all grammatical features are represented as POS tags.

Let us have the sets: LA — a set of POS tags[1], LE — a set of lemmas, WF — a set of word forms, and a set of dependency tags D ($ROOT \in D$). Let us have a sentence $x = w_1, ..., w_n$. A **tagged dependency tree** is a directed tree $T = (V, A, \pi, \lambda, \omega, \delta)$ where:

1. $V = \{0, 1, ..., n\}$ is an ordered set of nodes that corresponds to an enumeration of the words in the sentence (the root of the tree has index 0);

2. $A \subseteq V \times V$ is a set of arcs. For each node i, $1 \leq i \leq n$, there is exactly one arc in A: $\langle i, j\rangle \in A$, $0 \leq j \leq n$, $i \neq j$. There is exactly one arc $\langle i, 0\rangle \in A$;

3. $\pi : V - \{0\} \rightarrow LA$ is a total labelling function from nodes to POS tags[2]. π is not defined for the root;

4. $\lambda : V - \{0\} \rightarrow LE$ is a total labelling function from nodes to lemmas. λ is not defined for the root;

5. $\omega : V - \{0\} \rightarrow WF$ is a total labelling function from nodes to word forms. ω is not defined for the root;

6. $\delta : A \rightarrow D$ is a total labelling function for arcs. Only the arc $\langle i, 0\rangle$ is mapped to the label $ROOT$;

7. 0 is the root of the tree.

[1] In the formal definitions here we use tags as entities, but in practice they are sets of grammatical features

[2] In case when we are interested in part of the grammatical features encoded in a POS tag we could consider p as a set of different mappings for the different grammatical features. It is easy to extend the definition in this respect, but we do not do this here.

We will hereafter refer to this structure as a parse tree for the sentence x. The node 0 does not correspond to a word form in the sentence, but plays the role of a root of the tree.

Let $T = (V, A, \pi, \lambda, \omega, \delta)$ be a tagged dependency tree.

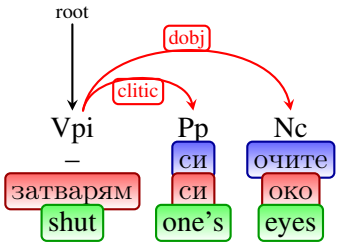
Let $T = (V, A, \pi, \lambda, \omega, \delta)$ be a tagged dependency tree. A directed tree $G = (V_G, A_G, \pi_G, \lambda_G, \omega_G, \delta_G)$ is called **dependency catena of** T if and only if there exists a mapping $\psi : V_G \to V$[3] such that:

1. $A_G \subseteq A$, the set of arcs of G;
2. $\pi_G \subseteq \pi$ is a partial labelling function from nodes of G to POS tags;
3. $\lambda_G \subseteq \lambda$ is a partial labelling function from nodes to lemmas;
4. $\omega_G \subseteq \omega$ is a partial labelling function from nodes to word forms;
5. $\delta_G \subseteq \delta$ is a partial labelling function for arcs.

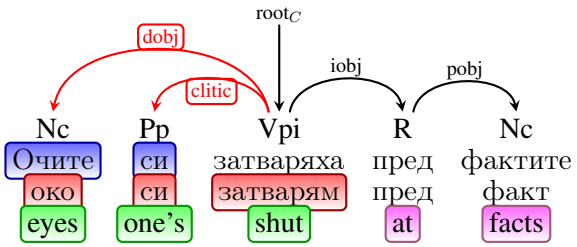
A directed tree $G = (V_G, A_G, \pi_G, \lambda_G, \omega_G, \delta_G)$ is a **dependency catena** if and only if there exists a dependency tree T such that G is a dependency catena of T.

Having partial functions for assigning POS tags, dependency labels, word forms and lemmas allows us to construct arbitrary abstractions over the structure of a catena. Thus, the catena could be underspecified for some of the node labels, like grammatical features, lemmas and also some dependency labels. The mapping ψ parameterizes the catena with respect to different dependency trees. Using the mapping, there is a possibility to realize different word orders of the catena nodes, for instance. The omission of node 0 from the range of the mapping ψ excludes the external root of the tagged dependency tree from each catena. CatR is the root of the catena. The catena could be a word or an arbitrary subtree.

We call the mapping of a catena into a given dependency tree the **realization of the catena in the tree**. We consider the realization of the catena as a fully specified subtree including all node and arc labels. For example, the catena for "to spill the beans" will allow for any realization of the verb form like in: "they spilled the beans" and "he spills the beans". Thus, the catena in the lexicon will be underspecified with respect to the grammatical features and word form for the verb.



Realization 1:



Realization 2:

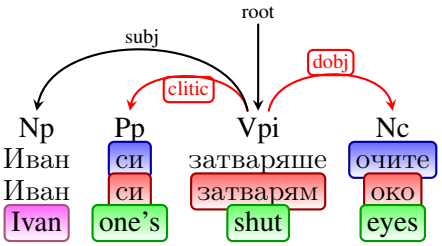


Figure 2: Catena realization

Sometimes this underspecified catena will be called a **lexicon catena** (LC), because its kind will be stored in the lexical entries. The Fig. 2 depicts two realizations (with different word orders) of the catena for the idiom затварям си очите ('zatvaryam si ochite', lit. shut one's eyes). The upper part of the image represents the lexicon catena for the idiom. It determines the fixed elements of the catena: the arcs, their labels, nodes and their labels: extended part of speech (first row), word forms (second row), lemmas (third row), and gloss in English (fourth row)[4]. The dash (–) in the word form row means that the word form is not defined

[3]This mapping allows for embedding of G in different tagged dependency trees and thus different word order realizations of the catena nodes (corresponding to word forms in T). The mapping y is specific for G and T. It allows also the image of G in T not to be a subtree of T, but several subtrees of T. A special case is discussed below — partition and extension operations.

[4]In the next examples we will present only the important information, thus, some of these rows will be missing. In other cases new rows will be used to represent additional information.

for the verbal node. In the two realizations the fixed elements of the catena are represented as in the image of the catena. The word order in the two realizations is different. Thus, using catenae with different underspecified elements defines different levels of freedom of realization of the multiword expressions.

Two catenae G_1 and G_2 could have the same set of realizations. In this case, we will say that G_1 and G_2 are **equivalent**. Representing the nodes via paths in the dependency tree from the root to the corresponding node and imposing a linear order over this representation of nodes facilitates the selection of a unique representative of each equivalent class of catenae. Thus, in the rest of the paper we assume that each catena is representative for its class of equivalence.

Let G_1 and G_2 be two catenae. A **composition** of G_1 and G_2 is a catena G_c, such that the catenae G_1 and G_2 are realized in G_e in such a way that the root node of G_2 is mapped to a node in G_c to which a node of G_1 is mapped. Each node in G_c is an image of a node from G_1 or G_2. The realizations of both catenae G_1 or G_2 share exactly one node in G_c. This node has to represent all the information from the nodes that are mapped to it. In this way we could realize the selectional restriction of a given lexical unit with respect to a catena in a sentence. For example, let us assume that the verb 'to read' requires a subject to be a human and an object to be an information object. In Fig. 3 we present how the catena for 'I read' is combined with the catena 'a book' in order to form the catena 'I read a book'. The figure represents only the level of word forms and a level of semantics (specified only for the node, on which the composition is performed). The catena for 'I read ...' specifies that the unknown direct object has the semantics of an *Information Object (InfObj)*. The catena for 'a book' represent the fact that the book is an Information Object. Thus the two catenae could be composed on the two nodes marked as InfObj. The result is represented at the bottom of the picture.[5]

Some MWEs require more complex operations over catenae in order to deal with them. Such a class of MWEs are idioms with an explicit subject, such as "the devil is in the details"; the realizations of catenae from the lexicon into syntax often are

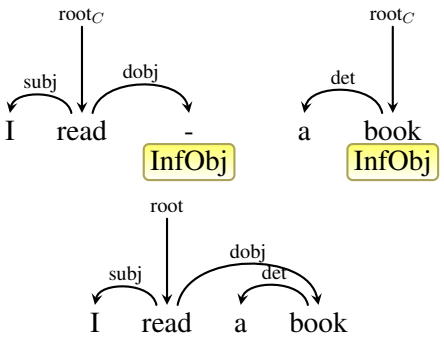


Figure 3: Composition of catenae.

accompanied by intervening material — see the discussion in (Osborne et al., 2012). For example, the idiom allows realizations such as: "the devil will be in the details", "the devil seems to be in the details", etc.

Our insight, supported by the examples, is that the intervening material forms a catena of a certain type. Such a type of catena will be called an **auxiliary catena**[6] in this paper, although it could be of different kinds (auxiliary, modal, control, etc.), depending on the verb forms. In order to implement this idea we need some additional notions.

Let $G = (V_G, A_G, \pi_G, \lambda_G, \omega_G, \delta_G)$ be a catena and $n \in V_G$, then $G_1, G_2, ..., G_n$ is a partition of G on node n if and only if for each $1 \leq i \leq n$:

1. each G_i is a catena which is a subtree of G
2. at most one subcatena G_i has n as a leaf node
3. one or more subcatenae G_i have n as a root node
4. the only common node for all subcatenae G_i is n
5. the mappings $\pi_{Gi}, \lambda_{Gi}, \omega_{Gi}, \beta_{Gi}$ are the same as for the whole catena G, except for the node n where the mappings $\pi_{Gi}, \lambda_{Gi}, \omega_{Gi}$ could be partial with respect to the original mappings.

An example of the operation **partition** of *the devil is in the details* is given in Fig. 4.

[5]In this representation many details like lemmas and grammatical features are not presented because they are not important for the example.

[6]Under auxiliary catena we understand a catena that is part of the verbal complex and contains nodes for the auxiliary verbs. In the grammars for the different languages different kinds of catena could be defined on the basis of there role in the grammar. In this respect the definition of extension here is restricted to verbal complex, but easy could be adapted for other cases when necessary.

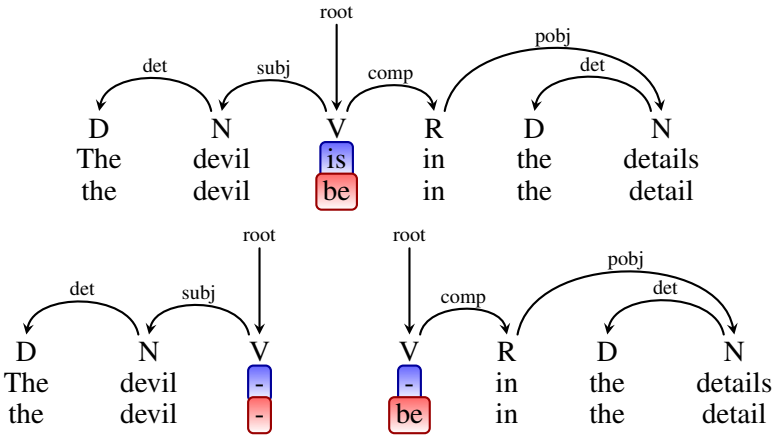


Figure 4: Partition

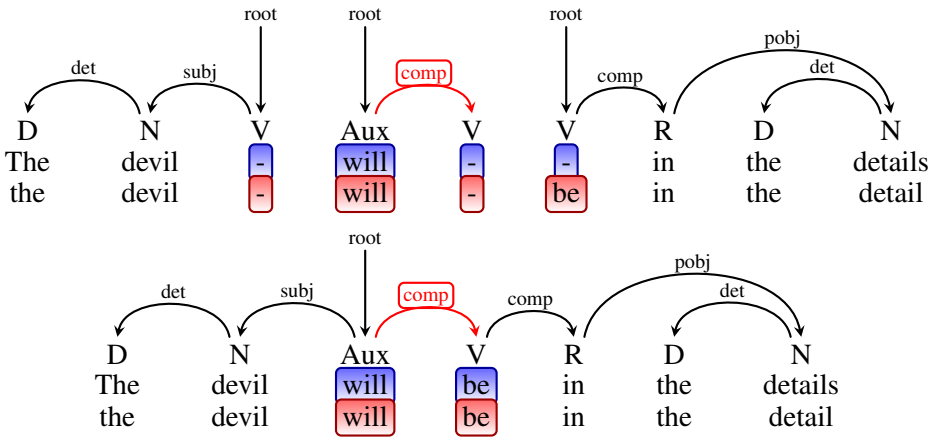


Figure 5: Extension

After the partition of a catena for an idiom we need a mechanism to connect the different catenae of the partition with the auxiliary catena.

Let G be a catena and for $n \in V_G, G_1, G, ..., G_n$ be a partition of G and G_a be an auxiliary catena. An **extension** of G on partition $G_1, G_2, ..., G_n$ with catena G_a is a catena G_e such that each catena $G_1, G_2, ..., G_n$ and the auxiliary catena G_a are realized in G_e in such a way that the node n_i in G_i (corresponding to the original node n) is mapped to a node in G_e to which a node of G_a is mapped. Each node in G_e is an image of a node from $G_1, G_2, ..., G_n$ or G_a.

An example of the operation **extention** is presented in Fig. 5[7]

Two catenae G_1 and G_2 could have the same set of realizations. In this case, we will say that G_1 and G_2 are **equivalent**. Representing the nodes via paths in the dependency tree from root to the corresponding node and imposing a linear order over this representation of nodes facilitates the selection of a unique representative of each equivalent class of catenae. Thus, in the rest of the paper we assume that each catena is representative of its class of equivalence. This representation of a catena will be called **canonical form**.

Using the notion of catena introduced in this section we define the structure of lexical items in the lexicon of a dependency grammar. Through the operations of composition, partition and extension we could define a procedure for analysis of actual sentences.

For each node in a catena or dependency tree we present the following information: POS, Grammatical Features, Word Form, Lemma, Node identifier (position of word form in a catena or a sentence). Each of the information is depicted in the node representation on a different row.

In order to model the behavior in a better way

[7]Notice that there are alternative analyses in which the auxiliary verb is not a head of the sentence, but a dependent of the copola.

we need to add semantics to the dependency representation. We will not be able to do this in full in this paper. In order to represent the interaction between lexical items and their valency frames in the lexicon, we assume a semantic analysis based on Minimal Recursion Semantics (MRS) (see (Copestake et al., 2005)). For dependency analyses, the MRS structures are constructed in a way similar to the one presented in (Simov and Osenova, 2011). In this work, the root of a subtree of a given dependency tree is associated with the MRS structure corresponding to the whole subtree. This means that for the semantic interpretation of MWEs we will use the root of the corresponding catena. In the dependency tree for the corresponding sentence the catena root will provide the interpretation of the MWE and its dependent elements, if any. In the lexicon we will provide the corresponding structure to model the idiosyncratic semantic content of MWE.

Our goal is to use catenae to represent the syntactic and morphological form of lexical units in the lexicon. The lexical units could be multiword expressions or single words. The lexical entry for a lexical unit has the following fields: **lexicon-catena** (LC) which contains a catena for the lexical item; **semantics** (SM) represents the semantic content of the lexical item; **valency frame** (Frame) contains a catena of the frame element and its semantics. The field Frame can be repeated as many times as necessary. Each valency frame corresponds to a syntactic relation of the dependent element. Alternative valencies for a given syntactic relation are represented in different Frame fields.

Here **lexicon-catena** determines the lexicon form of the lexical unit. The underspecification of the catena allows for the different realizations of the catena in the actual sentences. The **semantics** field defines the basic semantics of the lexical unit. The **valency frame** field provides selectional restriction for the lexical unit. Because the lexical unit could be a multiword expression, the semantics and selectional restrictions could be assigned to different nodes of the corresponding catena. In this way, different parts of the semantics could be provided by different nodes in the catena or from the catena related to the selectional restrictions. The selectional restrictions of a lexical unit also could be connected to different nodes of the lexical catena. In this way the lexical entry determines the possible variations of multiword expressions (MWEs). Below we will present concrete lexical entries for different types of lexical units, demonstrating selectional restrictions of verbs, nouns, multiword expressions.

4 Lexical Entry Examples

In this section we present some types of lexical entries using the structure of the lexical entry presented above. The examples are taken from the valency lexicon of Bulgarian, constructed on the basis of syntactic analyses, includes information about the main form (lemma) of the word, the valency frame with all the elements, their forms, grammatical features and semantics (Osenova et al., 2012). The lexical entry for each lexical item also includes the semantics of the main form and information on how this semantics incorporates the semantics of each frame element.

Here we first present the structure of the lexical entry for the verb ‘бягам’ (’byagam’, run) in the sense "run away from facts". The verb takes an indirect object in the form of a prepositional phrase starting with the preposition ‘от’ (’ot’, from). In the following examples we will omit the title row of the table for space reasons.

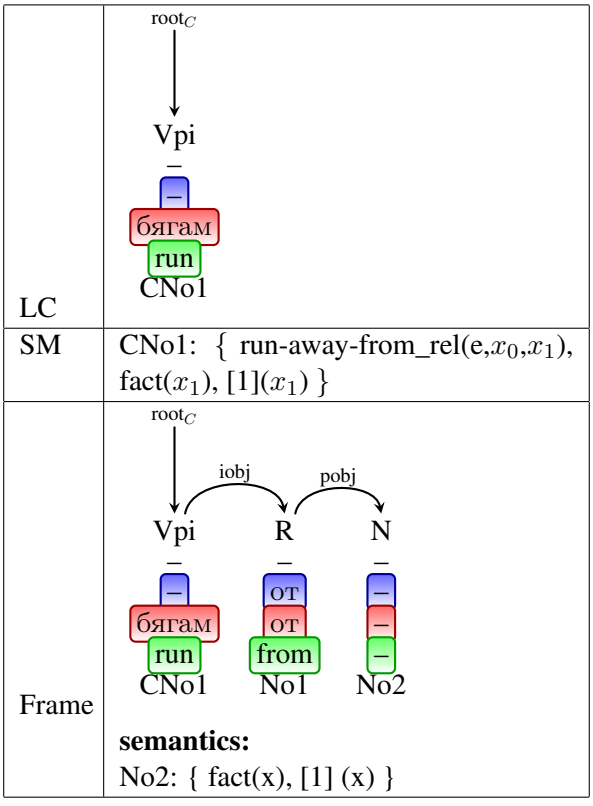


Figure 6: Lexical entry for the verb бягам "byagam", ‘run’)

In this model we use catenae for the representation of a single word and a MWE, because by definition single words are also catenae. Using the formal definition of catena from above, we might specify all grammatical features of the lexical item. The semantics in the lexical entry could be attached to each node in the lexicon-catena. In this example, there is just one node of the lexicon-catena. In the paper we present only the set of elementary predicates instead of the full MRS structures with the aim to demonstrate the principles of the representation. In the example, the verb introduces three elementary predicates: *run-away-from_rel*(e, x0, x1), *fact*(x1), [1](x1). The predicate *run-away-from_rel*(e, x0, x1) represents the event and its main participants: x0, x1. The predicate *fact*(x1) is part of the meaning of the verb in the sense that the agent represented by x0 will run away from some fact. There is also one underspecified predicate [1](x1) which has to be compatible with the predicate *fact*(x1). This predicate is used for incorporating the meaning of the indirect object. The valency frame is given as a set of valency elements. They are defined as a catena and semantic description. The catena describes the basic structure of the valency element including the necessary lexical information, grammatical features, the syntactic relation to the main lexical item. The semantic description determines the main semantic contribution of the frame element and via structural sharing it is incorporated in the semantics of the whole lexical item. In the example there is only one frame element. It is introduced via the preposition 'от' (from). The semantics comes from the dependent noun which has to be compatible with *fact*(x) predicate and via the underspecified predicate [1](x1) which could specify a more concrete predicate. Via the structure sharing index [1] this specific predicate is copied to the semantics of the main lexical item.

The lexical entry of a MWE uses the same format: a **lexicon-catena**, **semantics** and **valency**. The lexicon-catena for the MWEs is stored in its canonical form as described above. The semantics part of a lexical entry specifies the list of elementary predicates for the MRS analysis. When the MWE allows for some modification (also adjunction) of its elements, i.e. modifiers of a noun, the lexical entry in the lexicon needs to specify the role of these modifiers. For example, the MWE from the above example 'затварям си очите' which is synonymic to the verb 'byagam' presented above, is presented in Fig. 7[8]. The lexical entry is similar to the one shown earlier. The main differences are: the lexicon-catena is for the MWE instead of a single word. The semantics is the same, because the verb and the MWE are synonyms. The valency frame contains two alternative elements for indirect object introduced by two different prepositions. The situation that the two descriptions are alternatives follows from the fact that the verb has no more than one indirect object. If there is also a direct object then the valency set will contain elements for it as well.

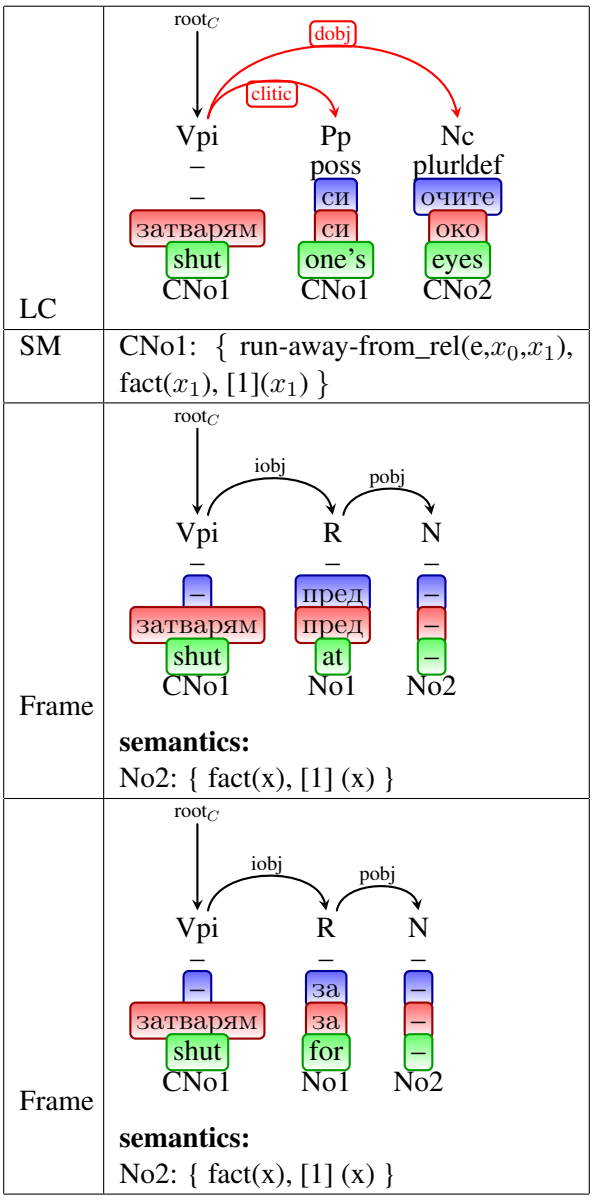


Figure 7: Lexical entry for затварям си очите "zatvaryam si ochite", 'I close my eyes'

[8]The grammatical features are: 'poss' for possessive pronoun, 'plur' for plural number and 'def' for definite noun.

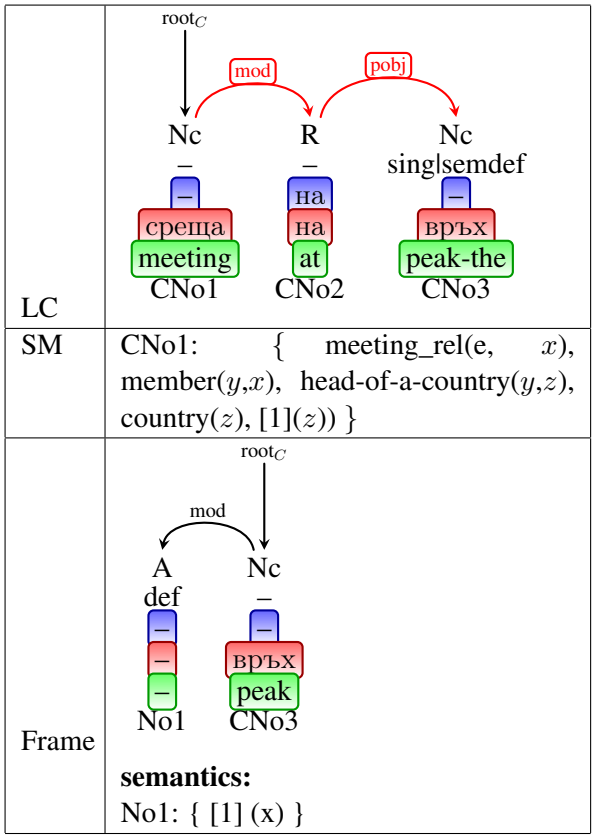


Figure 8: Lexical Entry for среща на върха “sresta na varha", ‘summit’

The semantics and the valency information are attached to the corresponding nodes in the catena representation. In the example in Fig. 7 only the information for the root node of the catena is given (identifier CNo1).

In cases when other parts of the catena allow modification, the information for the corresponding nodes will be given. Here we provide examples of such cases. For example, the Multiword Expression ‘среща на върха’ (’sreshta na varha’, summit) allows for modification not only of the whole catena, but also of the noun within the prepositional phrase. The lexical entry is given in Fig. 8[9]. This lexical entry allows modifications like ‘европейски’ (European) — среща на европейския връх (’sreshta na evropeyskiya vrah’, meeting of the European top). This catena allows also modification of the head word.

The next example presented here is for the multiword ‘снежен човек’ meaning “a man-like sculpture from snow”. It does not allow any modification of the dependent node ‘снежен’ (snowy),

[9]The grammatical features are: ‘sing’ for singular number and ‘semdef’ for definite subtree. Features like ‘semdef’ are specified for root node, but can be realized on a form inside the subtree.

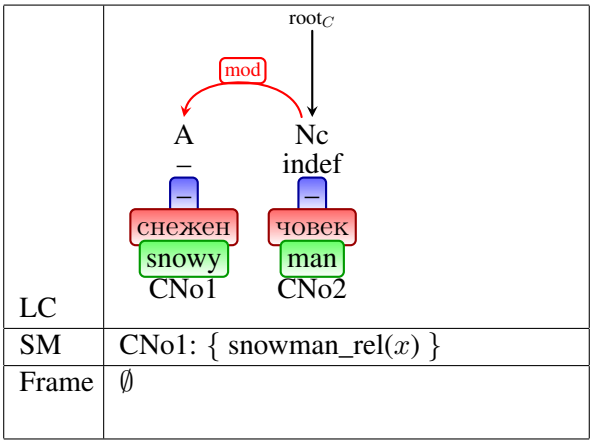


Figure 9: Lexical Entry for снежен човек “snezhen chovek”, ‘snowman’

but it allows for modifications of the root like “large snow man” etc. The lexical entry is given in Fig. 9[10]. The grammatical features for the head noun (indef for indefinite) restricts its possible form. In this way, singular and plural forms are allowed. The empty valency ensures that the dependent adjective cannot be modified except for morphological variants like singular and plural forms, but also definite or indefinite forms depending on the usage of the phrase. The possible modifiers of the MWE are determined by the represented semantics. The relation *snowman_rel*(x) is taken from an appropriate ontology where its conceptual definition is given.

Fig. 10 shows an example of non-verbal valency: the lexical entry of the relational noun ‘баща [на ...]’ (’bashta na...’, father of ...).

In the example so far, the selectional restrictions are potential and it is possible for them not to be realized in the actual text. But in some cases they are obligatory. Here we present one such example for the verb ‘състоя се’ (*’sastoya se ot’, consist of*). It requires an obligatory indirect object introduced by the preposition ‘от’ (’ot’, from) as in the sentence: Системата се състои от два модула (’Sistemata se sastoi ot dva modula’, The system consists of two modules.). In order to ensure that the indirect object will be always realized, we encode the preposition as an element of the lexicon catena. See the lexical entry in Fig. 11[11].

These examples demonstrate the power of the combination of catenae (as subtree units), MRS structures (as semantic units) and valency rep-

[10]The grammatical feature is: ‘indef’ for indefinite noun

[11]The grammatical feature is: ‘ref’ for reflexive pronoun

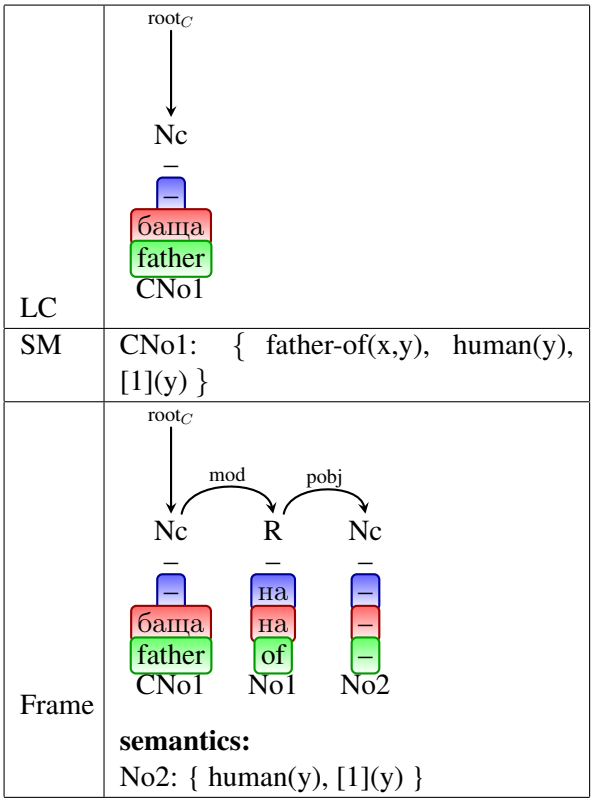


Figure 10: Lexical Entry for баща на "bashta na", 'father of'

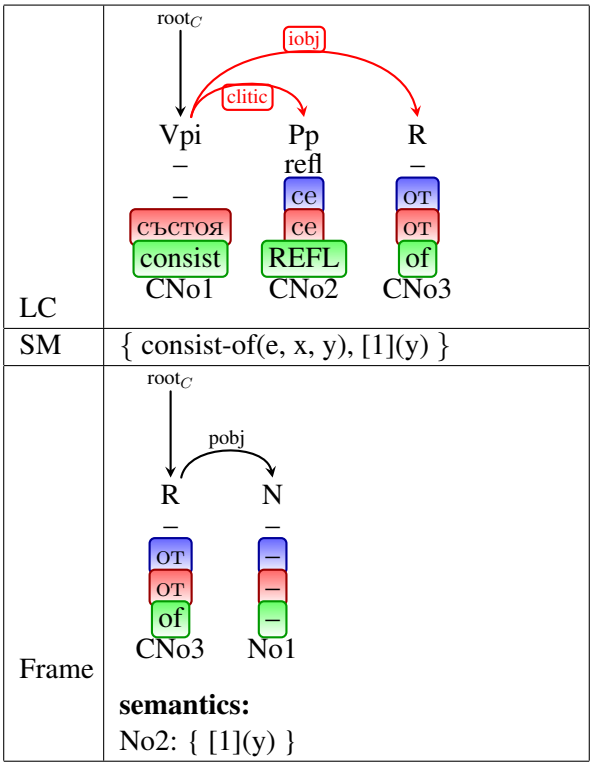


Figure 11: Lexical Entry for състоя се от "sastoya se ot", 'I consist of'

resentation (as subcategorization units) to model MWEs and valencies in the lexicon. The catena is appropriate for representation of syntactic structure; the semantic part represents the idiosyncratic semantics of the MWE and the semantics of valencies and determines the possible semantic modification, and the valency part determines the syntactic behavior of MWEs and other dependency expressions. One missing element of the lexical entry is the representation of constraints over the word order of the catena nodes. We envisage addition of such constraints as future work. The information from the lexical entries is combined by different operations on the elements of the lexical entries structure. The main operation on catenae is the realization in dependency trees. The two other operations are *extension* and *composition* of catenae. The *extension* is used when an MWE or other catena needs to be realized together with an auxiliary catena as in the case of sentence MWEs where the subject catena is detached from the verbal catena and realized as a subject of the auxiliary catena (see the example in Fig. 5). The *composition* is used when the valency catena is realized with the main lexical catena (see the example in Fig. 3).

5 Conclusion and Future Work

The paper demonstrates using Bulgarian data that the modeling at the level of catena is appropriate for encoding language units (including multiword expressions and valencies) at the lexicon-syntax interface. The catena allows for additional material to be inserted, based on the information from valence lexicons and contexts. Additionally, a semantics component is added for ensuring the correct interpretation of the language units.

The paper confirms the conclusions from previous works that catena is an appropriate means for encoding idioms and idiosyncratic language material. With respect to idioms it is very useful for cases where in addition to the figurative meaning the literal meaning also remains a possible interpretation. The paper also extends the catena mechanism to incorporate valency and semantic information.

The formalization of the catena provides definitions of operations over catenae which allow combination of catenae in complete analyses of sentences. In our work here we assume that catenae could have only one node in common — the node on which they extend or combine. This assumption is motivated by the examples of MWEs

that are idioms. Idioms usually interact with other catenae in a sentence via one of their nodes. But this requirement might be relaxed for the other catenae in the lexicon. In this way, in valency one could specify more than one common node between the lexical catena and the valency catena.

We do not employ any specific dependency theory in our approach, but we believe that the proposed modeling might be incorporated in most of them, if not all.

Acknowledgements

This research has received support by the EC's FP7 (FP7/2007-2013) under grant agreement number 610516: "QTLeap: Quality Translation by Deep Language Engineering Approaches" and by European COST Action IC1207: "PARSEME: PARSing and Multi-word Expressions. Towards linguistic precision and computational efficiency in natural language processing."

We are grateful to the three anonymous reviewers, whose remarks, comments, suggestions and encouragement helped us to improve the initial variant of the paper. All errors remain our own responsibility.

References

Ann Copestake, Dan Flickinger, Carl Pollard, and Ivan Sag. 2005. Minimal recursion semantics: An introduction. *Research on Language & Computation*, 3(4):281–332.

Thomas Gross. 2010. Chains in syntax and morphology. In Ryo Otoguro, Kiyoshi Ishikawa, Hiroshi Umemoto, Kei Yoshimoto, and Yasunari Harada, editors, *PACLIC*, pages 143–152. Institute for Digital Enhancement of Cognitive Development, Waseda University.

Marco Kuhlmann. 2010. *Dependency Structures and Lexicalized Grammars: An Algebraic Approach*, volume 6270 of *Lecture Notes in Computer Science*. Springer.

K. Tamsin Maxwell, Jon Oberlander, and W. Bruce Croft. 2013. Feature-based selection of dependency paths in ad hoc information retrieval. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 507–516, Sofia, Bulgaria, August. Association for Computational Linguistics.

William O'Grady. 1998. The syntax of idioms. *Natural Language and Linguistic Theory*, 16:279–312.

Timothy Osborne, Michael Putnam, and Thomas Groß. 2012. Catenae: Introducing a novel unit of syntactic analysis. *Syntax*, 15(4):354–396.

Petya Osenova, Kiril Simov, Laska Laskova, and Stanislava Kancheva. 2012. A Treebank-driven Creation of an OntoValence Verb Lexicon for Bulgarian. In *Nicoletta Calzolari (Conference Chair) and Khalid Choukri and Thierry Declerck and Mehmet Uğur Doğan and Bente Maegaard and Joseph Mariani and Jan Odijk and Stelios Piperidis (eds. Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, pages 2636–2640.

Chris Quirk, Arul Menezes, and Colin Cherry. 2005. Dependency treelet translation: Syntactically informed phrasal smt. In *Proceedings of ACL*. Association for Computational Linguistics, June.

Kiril Simov and Petya Osenova. 2011. Towards minimal recursion semantics over bulgarian dependency parsing. In *Proceedings of the RANLP 2011*.

Kiril Simov and Petya Osenova. 2014. Formalizing multiwords as catenae in a treebank and in a lexicon. In *Verena Henrich, Erhard Hinrichs, Daniël de Kok, Petya Osenova, Adam Przepiórkowski (eds.) Proceedings of the Thirteenth International Workshop on Treebanks and Linguistic Theories (TLT13)*, pages 198–207.