Modelling Derivational Morphology: A Case of Prefix Stacking in Russian

Yulia Zinova

Heinrich Heine University, Düsseldorf

Abstract. In order to automatically analyse Russian texts, one needs to model complex verb formation, as it is a productive mechanism and dictionary data is not sufficient. In this paper I discuss two implementations that aim to produce all and only the existing complex verbs built from the available morpheme inventory for the same fragment of Russian grammar. The first implementation is based on the syntactic theory approach to prefix combinatorics by Tatevosov (2009) and the other one uses the combination of basic syntactic restrictions and frame semantics to construct all possible combinations. I show that a combination of basic syntactic and semantic restrictions provides better results than a set of elaborated syntactic restrictions, especially for the complex verbs that are not normally tested by introspection.

1 Russian Verbal Prefixation System

Russian verbal derivational morphology is extremely rich. One stem can serve as a base for deriving hundreds of verbs via prefixation and suffixation. This is possible due to a high number of prefixes (28 according to Švedova 1982, p. 353, whereby most of them have productive usages), polysemy of prefixes (e.g., the prefix *pere-* has 10 usages according to Švedova 1982, pp. 363–364), and the possibility of prefix stacking. In addition to this, at some stages of the derivation (once per derivation) the imperfective suffix can be attached to the verb. The number and order of affixes, in turn, influences the aspect: prefixation usually leads to the perfective aspect of the derived verb and suffixation leads to the imperfective aspect of the derived verb. On the other hand, both prefixation and suffixation processes are restricted: not any prefix can be attached to a given verb (either simplex or complex), and the suffixation is not always available.

To show how the whole system functions together, let me provide an example. We start with a simplex verb *pisat'* 'to write'. It is imperfective and refers to an unbounded writing activity. If it is prefixed with *za-*, the derived verb *zapisat'* 'to record' is perfective and refers to a comlpeted event of recording something. It can be, in turn, suffixed, and the derived verb *zapisyvat'* 'to record/be recording' is imperfective. Yet another prefixation step can be made, for example with the prefix *do-*, which results in the derived perfective verb *dozapisyvat'* 'to finish recording again', as shown in (1).[1]

[1] IPF superscript marks the imperfective aspect of the verb and PF superscript marks the perfective aspect of the verb.

(1) pisat'^{IPF} → zapisat'^{PF} → zapisyvat'^{IPF} → dozapisyvat'^{PF}
'to write' → 'to record' → 'to (be) record(ing)' → 'to finish recording'

On the other hand, the order of the last two steps can be reversed: if the prefix *do-* is attached to the verb *zapisat'* 'to record', the derived verb *dozapisat'* 'to finish recording' is perfective and can be suffixed with *-iva-*, producing the imperfective verb *dozapisyvat'* 'to finish/be finishing recording', as shown in (2). As a result, the verb *dozapisyvat'* 'to finish/be finishing recording' can be obtained through two different derivations, one leading to the perfective (1) and the other (2) leading to the imperfective aspect of the derived verb.

(2) pisat'^{IPF} → zapisat'^{PF} → dozapisat'^{PF} →
'to write' → 'to record' → 'to finish recording' →
dozapisyvat'^{IPF}
'to be finishing recording'

To illustrate the limits of the derivational morphology, let us try to change the order of prefix attachment in the derivation (2). If the prefix *do-* is attached first, the derived verb *dopisat'* 'to finish writing' exists and is perfective. It is, however, not possible to attach the prefix *za-* to it: the verb **zadopisat'* does not exist (3).

(3) pisat'^{IPF} → dopisat'^{PF} *→ *zadopisat'
'to write' → 'to finish writing'

The goal of both accounts I discuss in this paper is to predict which complex verbs can be derived using the given set of morphemes and which aspect they will have. The implemented grammar fragment contains the following elements: a verb *pisat'* 'to write', a prefixed verb *zapisat'* 'to record', prefixes *po-* (delimitative and distributive interpretations), *pere-* (repetitive and distributive interpretations), and *do-* (completive interpretation), and the imperfective suffix *-iva-*. With this inventory I construct verbs with a maximum of four affixes (this can be realised if the base verb is prefixed two times, then suffixed, and then prefixed again).

Two alternative implementations proposed in this paper are based on two approaches to the prefixation system. In Section 2, I present the syntactic approach which is used as a base for the first implementation. Then, in Section 3, I introduce the frame semantic approach which motivates the second implementation. I then show that replacing complex syntactic restrictions with a combination of simple syntactic restrictions and semantic restrictions allows for better predictions of the existence and aspect of complex verbs with respect to both precision and recall.

2 Syntactic Approach

2.1 Theory

The main idea of approaches that pursue a syntactic view of the prefixation is to represent the verbal structure by means of a syntactic tree (Babko-Malaya 1999) and divide various prefix usages into categories such that each category is related to a specific position in the tree. This allows to restrict the available derivations. For example, according to this view, the derivation (3) is blocked because the prefix *za-* cannot occupy the position higher than that of the prefix *do-*, as the prefix *za-*, with the usage it has in the verb *zapisat'* 'to record', is classified as lexical whereas the prefix *do-* is classified as superlexical.

These two categories—lexical and superlexical prefixes—form a base of any existent syntactic approach to prefixation, as the two classes are claimed to exhibit distinct properties due to the different structural positions in the verbal tree. Such an approach is pursued by Svenonius (2004a,b), Romanova (2006), Ramchand (2004), Tatevosov (2007, 2009), among others. A serious problem of all these accounts is that they implicitly predict the non-existence of biaspectual verbs, as the highest affix in the structure serves to determine the aspect of the whole verb and there is only one possible structure for any verb with a fixed interpretation.[2]

In this paper I address the theory proposed by Tatevosov (2009) that stems from this line of research but includes substantial modifications with respect to the earlier proposals. Tatevosov (2009) divides the class of superlexical prefixes into three groups: selectionally limited, positionally limited, and left periphery prefixes.

Selectionally limited prefixes can be added only to a formally imperfective verb. The group includes the delimitative prefix *po-* (*posidet'* 'to sit for some time'), the cumulative prefix *na-*, the distributive prefix *pere-* (*perelovit' X* 'to catch all of X'), and the inchoative prefix *za-*. The group of positionally limited prefixes is constituted by the completive prefix *do-* (*dodelat'* 'to finish doing'), the repetitive prefix *pere-* (*perepisat'* 'to rewrite'), and the attenuative prefix *pod-*. These prefixes, according to Tatevosov (2009), can be added only before the secondary imperfective suffix *-yva-/-iva-*. The group of left periphery prefixes is constituted by only one prefix: distributive *po-* (*pobrosat'* 'to throw all of'). It occupies the left periphery of the verbal structure.

Such a division of superlexical prefixes into several subclasses allows Tatevosov (2009) to effectively limit the number of complex verbs. One drawback of the analysis is, as mentioned above, the prediction of the absence of complex biaspectual verbs (missing verb-aspect pairs). In order to better test the accuracy of the proposal by Tatevosov (2009), I have implemented it for the grammar fragment described above. In the next section I show fragments of the implementation and explain decisions that I had to make.

---

[2] In any existent syntactic analysis either (1) or (2) is not a valid derivation. For details, see (Zinova and Filip 2013).

2.2 Implementation

For the implementation, I have used EXtensible MetaGrammar[3] (XMG, Crabbé et al. 2013, Petitjean et al. 2016—a formalism that allows to describe linguistic information contained in the grammar and a tool to compute grammar rules and produce a redundant, strongly lexicalised Tree Adjoining Grammar (TAG, Joshi and Schabes 1997). In particular, the compiler for the current implementation is created using XMG 2 and has a syntactic (syn) and a frame semantic (frame) dimension (Lichte and Petitjean 2015). The code and the xml file that is output by the compiler are available online.[4]

The syntactic dimension is described using the following elements: first, all the nodes are declared using the keyword `node` and a variable name. These declarations are accompanied by optional marks (in brackets) and syntactic features (in square brackets, separated by commas). Values of syntactic features can be either specified or represented by a variable to ensure the same value of the feature across the nodes without specifying it. Second, the relations between the nodes are stated (immediate dominance, dominance, immediate precedence, precedence).

XMG is designed to output unanchored TAG elementary trees, but as currently there is no parser that would take into account the frame semantic dimension, I simulate the insertion of lexical anchors in the metagrammar. This solution leads to a more complicated metagrammar architecture, but allows us to see the results in a form that can be easily understood. If I were to output the unanchored trees only, I would obtain prefixation schemes but the stem that carries important information would not be inserted, which would make it very hard to check the predictions of the second implementation. (For the first implementation it would not make much difference as the only property of the verbal stem that can influence prefixation patterns in their productive part is aspect.)

As the first implemented approach is syntactic, all restrictions are formulated in syntactic terms and the frame dimension is used to represent the order of attachment of affixes with different semantics. For example, the class for the distributive interpretation of the prefix *pere-* looks as shown on Fig. 1. The restriction on this prefix attachment is the imperfective aspect of the base verb, which is reflected via a syntactic constraint on the feature *aspect* of the node `?VPInt`. The top node after the prefixation is `?VP` and it is characterized by the *perf* value of the feature *aspect*. The semantic dimension is a dummy for storing the relevant usage labels (as they are related to distinct syntactic properties) and keeping track of the approximate meaning of the derived verb so it can be compared with the exact meaning we will be dealing with in the second interpretation.

Other constraints (restricting the position of the prefix either to that below the imperfective suffix or to the leftmost slot in the structure) are realized through limiting the classes that can be assembled with the derivational base at each step.

[3] http://xmg.phil.hhu.de/

[4] https://user.phil-fak.uni-duesseldorf.de/ zinova/XMG/index.html

```
class PereVerb
export ?VP ?VPInt
declare ?VP ?VPInt ?Pere ?PereLex ?AGR ?X0 ?X1
{ <syn>{
    node ?VP [cat=vp, agr=?AGR, e=?X1, aspect = perf];
    node ?Pere [cat=pref];
    node ?PereLex (mark=flex) [cat=pere-];
    node ?VPInt [cat=vp, agr=?AGR, e=?X0, aspect = imperf];
    ?VP -> ?VPInt; ?VP -> ?Pere; ?Pere -> ?PereLex; ?Pere >> ?VPInt
  }; <frame>{
    ?X1[distributive,
       of: ?X0] } }
```

Fig. 1. XMG implementation for the distributive interpretation of the prefix *pere-* following Tatevosov (2009)

The output of this implementation consists of 81 models. Each model is associated with a certain verb with fixed order and interpretation of the prefixes. As an example, let me show the output for the verb *dozapisyvat'* 'to (be) finish(ing) recording' that we have discussed in Section 1. On Fig. 2, one can see that the last step of the derivation is the attachment of the imperfective suffix and the aspect of the derived verb is imperfective. This corresponds to the derivation (2). There is no other model in the output that would correspond to the derivation (1).

To evaluate the output, I have manually checked the existence of all the possible verb-aspect pairs using the assumption that an existent verb cannot be derived from a non-existent base. The total number of possible complex verbs built from the given morpheme inventory and paired with aspect is 546. The number of existent verb-aspect pairs that can be created out of this set, according to my count, is 70. Out of them the implementation of the account proposed by Tatevosov (2009) produces 52, which amounts to 0,642 precision and 0,743 recall for the implemented grammar fragment. In the next section I show how this result can be improved if complex syntactic restrictions are replaced with a combination of basic syntactic restrictions with semantic restrictions.

3 Alternative: Frame semantics

3.1 Framework

I argue that Russian verbal prefixation is a complex system that cannot be successfully modelled by means of one linguistic layer. In order to simplify individual components of the system and allow for the observed flexibility without massive overgeneration, one needs to coordinate the work of the morphological, syntactic, semantic, and pragmatic dimensions. In the fragment I describe here I limit myself to the first three systems, leaving pragmatic strengthening for future work.

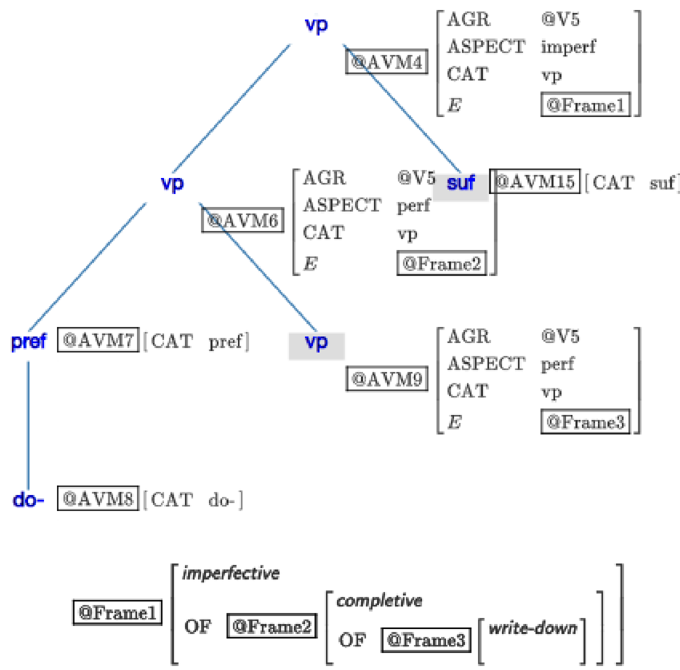

Fig. 2. XMG model for the verb *dozapisyvat'* 'to (be) finish(ing) recording'

Following Kallmeyer and Osswald (2012, 2013), I adopt a combination of frame semantics (Fillmore 1982) and Lexicalized Tree Adjoining Grammars (LTAG, Joshi and Schabes 1997, Frank 1992, Abeillé and Rambow 2000, Frank 2002). This framework has various benefits, such as a transparent syntax-semantics interface, numerous factorisation possibilities within the lexicon (especially important for the modelling of derivational morphology), and cognitive motivation. More information about the advantages of frame-based LTAG semantics can be found in (Kallmeyer and Osswald 2013).

The idea of using frame representations in linguistic semantics and cognitive psychology has been put forward by Fillmore (1982) and Barsalou (1992), among others. A widescale realisation of this idea is the Berkeley FrameNet project (Fillmore et al. 2003). The main ideas that motivate the use of frames as a general semantic and conceptual representation format can be summarized as follows (cf. Löbner 2014):

- conceptual-semantic entities can be described by types and attributes;
- attributes are functional relations, i.e., each attribute assigns a unique value to its carrier;
- attribute values can be also characterized by types and attributes (recursion);
- attribute values may be connected by additional relational constraints (Barsalou 1992) such as spatial configurations or ordering relations.

These ideas are formalized by Kallmeyer and Osswald (2013) who define frames as finite relational structures in which attributes correspond to functional relations. The members of the underlying set are referred to as the *nodes* of the frame. An important restriction is that any frame must have a *functional backbone*. This means that every node has to be accessible via attributes from at least one of the *base nodes*: nodes that carry *base labels* (unique identifiers). Importantly, feature structures may have multiple base nodes. In such a case often some nodes that are accessible from different base nodes are connected by a relation.

Another important component of the formalism is the type hierarchy. Since the number of syntactic restrictions I use is very limited, many derivations will be filtered out by the semantic constraints. For this, there are two main mechanisms: unification failure (type incompatibility or conflicting attribute values) and constraint failure (requirement for the two values to be in a specific relation is not satisfied). It is important to note that in the formalisation of Frame Semantics proposed by Kallmeyer and Osswald (2013) all types are considered compatible unless stated otherwise, so all the type conflicts have to be listed explicitly.

3.2 Frame Representations for Selected Prefixes

In this section I show frame representations of prefixes that are included in the implemented grammar fragment: *po-, pere-, do-*. Due to the space constraints I present only the representations and skip the theoretical motivation for them.

The Prefix *po-* The first prefix I provide a frame representation for is *po-*. The usages that are of interest for the implementation are the delimitative (*posidet'* 'to sit for some time') and the distributive (*pobrosat'* 'to throw all of') ones. On the basis of the discussions in (Filip 2000, Kagan 2015) and (Zinova 2017, Chapter 4), I propose to represent the contribution of the prefix by the frame shown on the left side of Fig. 3. The idea behind this representation is that the prefix adds information that the event is bounded (type *bounded-event*) and the initial (INIT) and the final (FIN) stages of the event are related to arbitrary points on the scale ($\boxed{2}$ and $\boxed{3}$ are free variables).

Such a representation allows the derivation of both delimitative and distributive usages of the prefix when the appropriate scale is selected. The equivalence of VERB-DIM and M-DIM attributes means that the appropriate scale must be equivalent to the verbal dimension. For the verb *pisat'* 'to write' this would be the time dimension that is realised as self-scaling. When this dimension is selected, the delimitative interpretation is acquired. If there is a source of iteration (e.g., a quantified object), the type of the M-DIM gets conjuncted with *cardinality* and the derived verb is interpreted distributively.

The Prefix *pere-* The prefix *pere-* is extremely polysemous. In this paper I consider two of its usages that are relevant for the implemented grammar

$$\mathbf{e}\begin{bmatrix} \textit{bounded-event} & \\ \text{VERB-DIM} & \boxed{1} \\ \text{M-DIM} & \boxed{1}\begin{bmatrix}\textit{scale}\end{bmatrix} \\ \text{INIT} & \begin{bmatrix} \textit{stage} & \\ \text{DEG} & \boxed{2} \end{bmatrix} \\ \text{FIN} & \begin{bmatrix} \textit{stage} & \\ \text{DEG} & \boxed{3} \end{bmatrix} \end{bmatrix} \qquad \mathbf{e}\begin{bmatrix} \textit{bounded-event} & \\ \text{NOUN-DIM} & \boxed{3} \\ \text{M-DIM} & \boxed{3}\begin{bmatrix} \textit{closed-scale} \wedge & \\ \textit{proper-scale} & \\ \text{MIN} & \boxed{1} \\ \text{MAX} & \boxed{2} \end{bmatrix} \\ \text{INIT} & \begin{bmatrix} \textit{stage} & \\ \text{DEG} & \boxed{1} \end{bmatrix} \\ \text{FIN} & \begin{bmatrix} \textit{stage} & \\ \text{DEG} & \boxed{2} \end{bmatrix} \end{bmatrix}$$

Fig. 3. Frame representations of the prefixes *po-* (left) and *pere-* (case of a closed scale, right). and frame for coercion of an unbounded event into a bounded event (right)

fragment: distributive (*perelovit' X* 'to catch all of X') and repetitive (*perepisat'* 'to rewrite').

The frame for the distributive usage of the prefix *pere-* is shown on the right side of Fig. 3. The key restrictive factor in this case is the type of the measure dimension (M-DIM) that has to be supplied by the context (M-DIM = NOUN-DIM, the context-determined dimension is called NOUN-DIM as usually it is the direct object that supplies it): a closed proper scale in this case (*closed-scale* ∧ *proper-scale*). The initial and final stages of the event correspond to the minimum and maximum points on the scale (INIT.DEG = M-DIM.MIN; FIN.DEG = M-DIM.MAX).

The repetitive usage of the prefix *pere-* arises when the measure dimension of the event denoted by the derivational base (PREP.M-DIM) is of type *property-scale* (Fig. 4, the types *proper-scale* and *property-scale* are not compatible with each other). This event then becomes a value of the preparatory phase (PREP) attribute of the new event. The initial and the final stages, the noun dimension, the measure dimension, and the manner attributes are copied to the event node that refers to the new event (M-DIM = PREP.M-DIM, FIN = PREP.FIN, INIT = PREP.INIT, MANNER = PREP.MANNER). The tree on the right side of 4 shows that the derived verb refers to a frame node (feature E, E = **f**) other than the derivational base (E = **e**). It also stores information about the noun to the right of the verb being the THEME of the original event (PREP.THEME = $\boxed{6}$, I = $\boxed{6}$).

The next restriction for the repetitive usage of the prefix *pere-*, apart from the property type of the scale, is that the event denoted by the derivational base must have a final stage in its representation. This means that a simplex imperfective verb cannot be combined with this prefix usage, unless it is coerced into a bounded event. On the formal side it means formulating a requirement on the frame configuration (the presence of the FIN attribute). For implementing the coercion of an unbounded event into a bounded event, I propose to use the frame shown on Fig. 5. It functions similarly to the frame for the distributive usage of

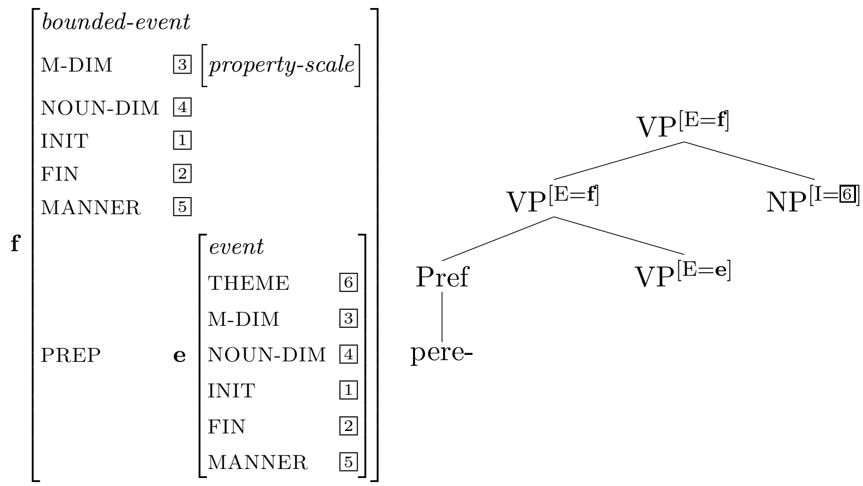

Fig. 4. Representation of the contribution of the prefix *pere-*: case of a property scale

the prefix *pere-* (on the right side of Fig. 3), varying from it only with respect to the type of the measure dimension (*property-scale* instead of *proper-scale*).

The Prefix *do-* The last prefix included in the implemented grammar fragment is the prefix *do-* with completive semantics. The event denoted by a *do*-prefixed verb is a terminal part of the original event. In other words, when the prefix is attached, the maximum of the scale has to be associated with the final stage of the event (M-DIM.MAX = FIN.DEG). The frame shown on Fig. 6 realizes this as well as the *do*-specific mechanism of scale selection (it can be either NOUN-DIM or VERB-DIM, but the types of all the dimensions have to be copied from the representation of the event denoted by the derivational base to the representation of the event denoted by the derived verb). Note that attributes in Frame Semantics are functional, so the attribute PART-OF has to satisfy this restriction as well. To ensure this, I define the value of this attribute as the maximum event that has the event in question as a part. In particular, it would be an event that proceeds from the minimum (**f**.M-DIM.MIN, $\boxed{3}$ in the frame above) to the maximum degree (**f**.M-DIM.MAX, $\boxed{1}$ in the frame above) on the relevant scale. The scale has to be closed in order for the value of the PART-OF attribute to be defined using the frame on the left side of Fig. 6.

Similarly to the iterative usage of the prefix *pere-*, the prefix *do-* can be only attached to bounded events. This means that, again, simplex imperfective verbs need to be first coerced into a bounded interpretation.

3.3 Implementation

Restrictions In this section I present the implementation of the frame-based proposal for limiting the derivation of complex verbs and predicting their aspect

$$\mathbf{e}\left[\begin{array}{ll} \textit{bounded-event} & \\ \text{NOUN-DIM} & \boxed{3} \\ \text{M-DIM} & \boxed{3}\left[\begin{array}{ll} \textit{closed-scale} \wedge & \\ \textit{property-scale} & \\ \text{MIN} & \boxed{1} \\ \text{MAX} & \boxed{2}\end{array}\right] \\ \text{INIT} & \left[\begin{array}{ll}\textit{stage} & \\ \text{DEG} & \boxed{1}\end{array}\right] \\ \text{FIN} & \left[\begin{array}{ll}\textit{stage} & \\ \text{DEG} & \boxed{2}\end{array}\right]\end{array}\right]$$

Fig. 5. Frame for coercion of an unbounded event into a bounded event

and semantics. As the frame domain is a new development in XMG, I had to deal with parser restrictions and place some semantically motivated constraints on the syntactic level.

First, I had to 'lift' two features (*bounded* and *limited*) to the syntactic level due to the fact that such feature checking inside the semantic dimension of XMG is not yet supported. The feature *bounded* appears at those nodes that are associated with frames of *event* type. It gets the value *yes* if there is a path from the central node of the frame to an attribute FIN that can proceed through the PART-OF attributes. This corresponds to the event being of type *bounded-event* or being a PART-OF an event of a *bounded-event* type. If there is no such path, the value of the feature is *no*. The feature *limited* is a stronger version of a similar constraint: for *limited* to get the value *yes*, the central node has to have an attribute FIN and its value has to be specific (concrete value or a bound variable). This corresponds to the event being of type *bounded-event*. In all other cases the feature *limited* gets the value *no*.

Another restriction that is located in the syntactic domain despite its semantic nature is the *unicity of iteration*. The idea is that inside a derivation there can be only one semantic marker of iteration. In the current implementation, this feature is doubled on the syntactic side because there is no possibility to check whether this constraint holds on the semantic side.

Type Hierarchy As I have noted above, type incompatibility is one of the mechanisms that blocks derivations. The type hierarchy description consists of two types of statements: (1) statements declaring that one type is also some other type, e.g. `property-scale -> scale` (something of type *property-scale* is also of type *scale*) and (2) statements declaring that types are not compatible, e.g. `cardinality property-scale -> -` (something of type *cardinality* cannot simultaneously be of type *property-scale*). In the implementation proposed here I postulate a minimum set of constraints that is sufficient to block unwanted

$$\mathbf{f}\begin{bmatrix} \textit{bounded-event} & \\ \text{MANNER} & \boxed{4} \\ \text{ACTOR} & \boxed{5} \\ \text{THEME} & \boxed{9} \\ \text{M-DIM} & \begin{bmatrix} \textit{closed-scale} \\ \quad \wedge\ \textit{property-scale} \\ \text{MIN}\ \boxed{2} \\ \text{MAX}\ \boxed{1} \end{bmatrix} \\ \text{INIT} & \begin{bmatrix} \textit{stage} \\ \text{DEG}\ \boxed{3} \end{bmatrix} \\ \text{FIN} & \begin{bmatrix} \textit{stage} \\ \text{DEG}\ \boxed{1} \end{bmatrix} \\ \text{PART-OF} & \mathbf{e} \\ \text{NOUN-DIM} & \begin{bmatrix} \boxed{7} \end{bmatrix} \\ \text{VERB-DIM} & \begin{bmatrix} \boxed{8} \end{bmatrix} \end{bmatrix} \qquad \mathbf{e}\begin{bmatrix} \textit{bounded-event} & \\ \text{MANNER} & \boxed{4} \\ \text{ACTOR} & \boxed{6} \\ \text{THEME} & \boxed{9} \\ \text{M-DIM} & \begin{bmatrix} \textit{closed-scale} \\ \quad \wedge\ \textit{property-scale} \\ \text{MAX}\ \boxed{1} \end{bmatrix} \\ \text{INIT} & \begin{bmatrix} \textit{stage} \\ \text{DEG}\ \boxed{2} \end{bmatrix} \\ \text{FIN} & \begin{bmatrix} \textit{stage} \\ \text{DEG}\ \boxed{1} \end{bmatrix} \\ \text{NOUN-DIM} & \begin{bmatrix} \boxed{7} \end{bmatrix} \\ \text{VERB-DIM} & \begin{bmatrix} \boxed{8} \end{bmatrix} \end{bmatrix}$$

$$\langle \mathbf{f} \cdot \text{M-DIM}, \mathbf{e} \cdot \text{M-DIM} \rangle : \textit{segm-of}$$

Fig. 6. Frame representation of the prefix *do-*

prefix combinations and is motivated by the internal structure of the scales in question.

Lexical Anchors In a proper implementation that would separate the metagrammar level from the syntactic level the following elements would not belong to the metagrammar, but would be used as lexical anchors for the appropriate tree families. The first entry is the noun that is used to fill the object slot. For the implementation proposed here I have selected the plural form of the noun *rasskaz* 'story'. For our purposes it is important that stories have length and (due to the plurality of the noun) also some cardinality.

The description of the noun (Fig. 7) is straightforward: on the syntactic side, it is a daughter of the N category node and on the semantic side it contains relevant attributes. The two nodes (?N and ?Story) are declared in the first two lines of the syntactic domain description and connected via an immediate dominance relation in the third line.[5] Both nodes are characterized with feature `i=?X0` which connects them to the semantic frame characterized in the frame dimension. The frame description states that the type of the frame ?X0 is `story` and it has two attributes: length and cardinality.

These attributes enable the noun to enter one of the dimension constructors. The cardinality constructor is available for all nouns that have a cardinality

[5] This unary branching is necessary in the current implementation due to the absence of a syntactic compiler that would work with frames: it models the lexical anchor insertion inside the metagrammar.

```
class Story
export ?Length ?Card ?N
declare ?N ?Story ?X0 ?Length ?Card
{ <syn>{
    node ?N (mark=coanchor) [cat=n, num = pl, i=?X0];
    node ?Story (mark=nounacc) [cat=rasskazy, num = pl, i=?X0];
    ?N -> ?Story
  }; <frame>{
    ?X0[story,
        length: ?Length,
        cardinality: ?Card ] } }
```

Fig. 7. XMG code: noun that is used to fill the accusative NP slot

attribute with an additional restriction for plural number. On the semantic side it creates an M-DIM attribute and the event description bounded to the VP node also acquires the type *iteration*. Another dimension constructor that I use, implemented in the class NounLength, creates a NOUN-DIM of type *property-scale* $\wedge$ *length* and the MAX being equal to the value of the LENGTH attribute of the noun.

The second group of lexical items consists of two verbs: *pisat'* 'to write' and *zapisat'* 'to write down'. The second verb contains the prefix *za-*, but its semantic contribution is not transparent (in terms of a syntactic approach it is a lexical prefix usage), so the whole verb must be stored in the dictionary. The class that represents the verb *pisat'* 'to write' has a simple syntactic structure of two nodes (see Fig. 8): the node of category V and the node that contains the verb itself, where the V node inherits all syntactic properties of the verb, except for the category. The *aspect* feature, in contrast with the features *limited* and *bounded*, is a syntactic feature and carries information about the syntactic aspect of the verb represented by the respective node. For the frame semantic side, I use a simple representation that serves the purposes of the current analysis.

Prefixes As we have already discussed the frames for all individual prefix usages in Section 3.2, I will now show how frames correspond to the XMG descriptions and what happens on the syntactic side, taking one prefix as an example.

Figure 9 shows the code for the prefix *po-*. The syntactic part of it represents a VP that consists of a prefix head and another (internal) VP that carries information about the derivational base. The agreement information as well as the semantic frame are then passed to the higher VP node. This node is also characterized by the `perf` (perfective) value of the aspect feature independently of the value of the aspect feature of the internal VP node. Following the definitions provided above, the feature *limited* is assigned the value *yes* because the semantic frame contains the attribute FIN, but the feature *bounded* is assigned the value *no*, as the value of the attribute FIN is a free variable.

```
class Pisat
export ?V
declare ?V ?Pisat ?X0 ?Actor ?Theme ?Mean
{ <syn>{
    node ?V (mark=anchor) [cat=v, e=?X0, asp = unbound, aspect = imperf];
    node ?Pisat (mark=flex) [cat=pisat, e=?X0, asp = unbound,
     aspect = imperf];
    ?V -> ?Pisat
  }; <frame>{
    ?X0[event & process,
        actor:?Actor,
        theme:?Theme,
        mean:?Mean,
        manner:[write],
        verb-dim:?X0 ] } }
```

Fig. 8. XMG code for representation of the verb *pisat'* 'to write'

```
class PoVerb
export ?VP ?VPInt
declare ?VP ?VPInt ?Po ?PoLex ?AGR ?X0 ?Init ?Fin ?VDim
{ <syn>{
    node ?VP [cat=vp, agr=?AGR, e=?X0, limited = yes, bounded = no,
      aspect = perf];
    node ?Po [cat=pref];
    node ?PoLex (mark=flex) [cat=po-];
    node ?VPInt [cat=vp, agr=?AGR, e=?X0, bounded = no];
    ?VP -> ?VPInt; ?VP -> ?Po; ?Po -> ?PoLex; ?Po >> ?VPInt
  }; <frame>{
    ?X0[bounded-event,
       m-dim: ?VDim,
       verb-dim: ?VDim,
       init: [stage,
          scale-deg:?Init],
       fin: [stage,
          scale-deg:?Fin] ] } }
```

Fig. 9. XMG code for the class describing the prefix *po-*

As for the frame description part, it follows the proposed frame configuration straightforwardly. This is evident if one compares the code with the frame on the left side of Fig. 3 for the prefix *po-*.

Encoding of other prefix usages proceeds in a similar way: the syntactic part does not vary much from prefix to prefix and semantic descriptions can be directly obtained from the frame descriptions I have proposed in Section 3.2. The coersion that is sometimes required before the attachment of the prefix *pere-* or the prefix *do-* is realised by the class NDimCoercedVerb that transforms a non-bounded event into a bounded event using the nominal scale.

Imperfective Suffix I use two separate classes to produce two interpretations of secondary imperfective verbs: progressive and habitual. For the analysis that I propose it is important to distinguish between them when another prefix is attached after the suffixation, as these two interpretations have different semantic properties.

The habitual interpretation of the imperfective suffix (*IterVerb* class) produces an unlimited event that is a series of limited events. The NOUN-DIM of the new event necessarily is of type *cardinality* and does not need to correspond to the respective attribute of the derivational base. The verbal dimension is copied from the individual event level to the series level. This interpretation of the imperfective suffix is also associated with the introduction of the *iteration* type of the event and the respective syntactic feature.

The second interpretation of the imperfective suffix is progressive: on the semantic side I represent it as a creation of a new event that is a PART-OF the event denoted by the derivational base. Due to the PART-OF relation the new event remains limited. As relations are currently not implemented in XMG, for the sake of the implementation I use PART-OF as an attribute when representing the progressive interpretation of the imperfective suffix (class *ProgrVerb*), although it is not functional.

Assembling the Parts The last part of the code assembles the verbal phrases from the components described above. This is done by the classes *OneBasePrefixedVerb*, *VerbWithOnePrefix*, *TwoPrefixedVerb*, *TwoPrefixedSuffixedVerb*.[6]

The compilation of the code produces 88 models (in this case not verbs, but verbal phrases). As these models include two interpretations of the imperfective suffix whereas such distinction is not made in the analysis of Tatevosov (2009), before calculating precision and recall one has to remove 'duplicates': models that differ only with respect to suffix interpretations. This leaves 79 models of

[6] An additional operation of type matching has to be performed after the suffixed verbs are assembled, as in the current version of XMG type copying is performed not via creating a connection between two types (as it is done with attributes), but by copying the value that is there at the moment the operation is performed. To ensure that later type enrichments are copied to all the necessary locations, the class *TypeMatcher* identifies all types of the measure dimensions (M-DIM, NOUN-DIM, VERB-DIM) between the higher and the embedded frames.

which 70 are correct and amounts to 0,886 precision and full recall. Nine extra models have to be filtered out later by the pragmatic module, but most part of the work is done by the syntactic and semantic constraints shown above.

4 Discussion

In this paper I have proposed two implementations of analyses that aim to predict the existence and aspect of complex Russian verbs. I have shown that an analysis that is based exclusively on syntactic restrictions (postulating a division of prefixes into several groups) does worse both with respect to precision and recall than an analysis that uses both simple syntactic and semantic restrictions (for the implemented grammar fragment). The summary of precision/recall data is provided in Table 1.

analysis	precision	recall	F-measure
frame-based analysis	0,886	1	0.94
Tatevosov (2009)	0,642	0,743	0.689

Table 1. Precision, recall and F-measure for two implementations

It is interesting to note that the difference is not huge if one considers only verbs with one or two prefixes and an imperfective suffix added at the last step of the derivation: the number of errors stays close (two versus three within the implemented fragment) and both implementations have full recall with respect to this part of the grammar. The comparison becomes more interesting when we consider the most complex verbs created by the two implementations. The number of models produced here is close: 45 models according to the analysis by Tatevosov (2009) and 49 models in the implementation of the frame semantic analysis. The overlap of these sets constitutes, however, only 27 models. One can argue that such verbs are rare, but I consider them to be an opportunity to test the model, as fitting the theory to such complex cases is not feasible.

Another remark I want to add is that all nine incorrect models that are encountered in the output of the second implementation can be filtered out using basic pragmatic reasoning. Besides, the output of the analysis contains fully spelled-out semantic representations that are obtained compositionally and the interpretation of the prefix in a given position is derived and not stipulated.

In future work I plan to extend the implementation of the frame-based analysis to a larger language fragment and test the predictions of the theory using not only the corpus data and introspection, but setting up experiments to verify the existence of certain complex verbs and then build a database that could be used for future research.

References

Abeillé, A., Rambow, O.: Tree Adjoining Grammar: An Overview. In: Abeillé, A., Rambow, O. (eds.) Tree Adjoining Grammars: Formalisms, Linguistic Analyses and Processing, pp. 1–68. CSLI (2000)

Babko-Malaya, O.: Zero Morphology: A Study of Aspect, Argument Structure, and Case. Dissertation, Rutgers University (1999)

Barsalou, L.W.: Frames, concepts, and conceptual fields. In: Lehrer, A., Kittay, E.F. (eds.) Frames, Fields, and Contrasts, chap. 1, pp. 21–74. New Essays in Semantic and Lexical Organization, Lawrence Erlbaum Associates, Hillsdale, New Jersey (1992)

Crabbé, B., Duchier, D., Gardent, C., Le Roux, J., Parmentier, Y.: XMG: eXtensible MetaGrammar. Computational Linguistics 39(3), 591–629 (2013)

Filip, H.: The quantization puzzle. In: Events as Grammatical Objects, pp. 3–60. CSLI Press, Stanford (2000)

Fillmore, C.J.: Frame Semantics. In: Linguistics in the Morning Calm, pp. 111–137. Hanshin Publishing Co., Seoul (1982)

Fillmore, C.J., Johnson, C.R., Pertuck, M.R.L.: Background to FrameNet. International Journal of Lexicography 3(16), 235–250 (2003)

Frank, R.: Syntactic Locality and Tree Adjoining Grammar: Grammatical, Acquisition and Processing Perspectives. Ph.D. thesis, University of Pennsylvania (1992)

Frank, R.: Phrase structure composition and syntactic dependencies. MIT Press, Cambridge, MA (2002)

Joshi, A.K., Schabes, Y.: Tree-Adjoining Grammars. In: Rozenberg, G., Salomaa, A. (eds.) Handbook of Formal Languages, pp. 69–123. Springer, Berlin (1997)

Kagan, O.: Scalarity in the Verbal Domain: The Case of Verbal Prefixation in Russian. Cambridge University Press (2015)

Kallmeyer, L., Osswald, R.: A frame-based semantics of the dative alternation in lexicalized tree adjoining grammars (2012), submitted to Empirical Issues in Syntax and Semantics 9

Kallmeyer, L., Osswald, R.: Syntax-driven semantic frame composition in lexicalized tree adjoining grammars. Journal of Language Modelling 1(2), 267–330 (2013)

Lichte, T., Petitjean, S.: Implementing semantic frames as typed feature structures with xmg. Journal of Language Modelling 3(1), 185–228 (2015)

Löbner, S.: Evidence for frames from human language. In: Gamerschlag, T., Gerland, D., Osswald, R., Petersen, W. (eds.) Frames and Concept Types, pp. 23–67. No. 94 in Studies in Linguistics and Philosophy, Springer, Dordrecht (2014)

Petitjean, S., Duchier, D., Parmentier, Y.: XMG 2: Describing description languages. In: Logical Aspects of Computational Linguistics 2016 (2016)

Ramchand, G.: Time and the event: The semantics of Russian prefixes. Nordlyd 32(2) (2004)

Romanova, E.: Constructing Perfectivity in Russian. Ph.D. thesis, University of Tromsø (2006)

Svenonius, P.: Slavic prefixes and morphology. An introduction to the Nordlyd volume. Nordlyd 32(2), 177–204 (2004)

Svenonius, P.: Slavic prefixes inside and outside VP. Nordlyd 32(2), 205–253 (2004)

Tatevosov, S.: Intermediate prefixes in Russian. In: Proceedings of the annual workshop on formal approaches to Slavic linguistics. vol. 16 (2007)

Tatevosov, S.: Množestvennaja prefiksacija i anatomija russkogo glagola [Multiple prefixation and the anatomy of Russian verb]. In: Kisseleva, X., Plungian, V.,

Rakhilina, E., Tatevosov, S. (eds.) Korpusnye issledovanija po russkoj grammatike [Corpus-based Studies in the Grammar of Russian], pp. 92–156. Probel, Moscow (2009)
Švedova, N.J.: Russkaja Grammatika, vol. 1. Nauka, Moscow (1982)
Zinova, Y.: Russian verbal prefixation. Ph.D. thesis, Heinrich-Heine University, Düsseldorf (2016)
Zinova, Y., Filip, H.: Biaspectual Verbs: A Marginal Category? In: Aher, M., Hole, D., Jeřábek, E., Kupke, C. (eds.) 10th International Tbilisi Symposium on Logic, Language, and Computation, pp. 310–332. Springer, Gudauri, Georgia (2013), paper presented at TbiLLC 2013