Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020), pages 3937–3943
Marseille, 11–16 May 2020
© European Language Resources Association (ELRA), licensed under CC-BY-NC

DerivBase.Ru: a Derivational Morphology Resource for Russian

Daniil Vodolazsky
Novosibirsk State University
Novosibirsk, Russia
daniil.vodolazsky@mail.ru

Abstract

Russian morphology has been studied for decades, but there is still no large high coverage resource that contains the derivational families (groups of words that share the same root) of Russian words. The number of words used in different areas of the language grows rapidly, thus the human-made dictionaries published long time ago cannot cover the neologisms and the domain-specific lexicons. To fill such resource gap, we have developed a rule-based framework for deriving words and we applied it to build a derivational morphology resource named DerivBase.Ru, which we introduce in this paper.

Keywords: derivational morphology, lexical resources, word graphs

1. Introduction

Morphological derivation is a process of producing new words. E.g., a word *worker* is derived from a word *work* with derivational suffix *-er*. In many NLP tasks, one often needs to find the words that are derivationally related (i.e. words that share the same root), for instance in paraphrase or plagiarism detection. This can be done with a derivational morphology resource. For a widely used Russian language such resources exist[1], but they are hand-made and limited with small number of words, and thus cannot be easily applied to the texts with new terminology. Tikhonov (1985) dictionary contains 145 000 words, but it is not available online, and its shorter version (Tikhonov, 2017) is not convenient for computational purposes.

Our work is inspired by Zeller et al. (2013) and extends the proposed approach to Russian language. DErivBase is a large-coverage derivational lexicon for German which consists of derivational families, groups of lemmas which are derivationally related among each other.

Russian is a fusional language, and it therefore presents a lot of prefixes and suffixes (both inflectional and derivational). The most productive ways of word formation in Russian are suffixation: *mechta (a dream) → mechtatel' (a dreamer)*, prefixation: *krichat' (to scream) → zakrichat' (to start screaming)*, circumfixation: *slepoy (blind) → [igrat'] vslepuyu ([playing] blind)*, compounding: *pyl' (dust), sosat' (to suck) → pylesos (a vacuum cleaner)*, and abbreviation: *matematicheskiy fakultet (mathematical department) → matfak*. In this work, we are focusing only on affixation, though our approach can be extended to other word formation types.

Given that not all the possible outputs of a derivational rule are realised in the language, we additionally use information about an *inflectional type* (Zaliznyak, 1980) of a source word (for nouns, adjectives and verbs), as it helps us to design rules more accurately. Every inflectional type represents a special pattern that is common to all words of a given class. E. g., consider a verb *um'e-t'* (can) and its wordforms *um'e-ju* (1per, sg), *um'e-jet* (3per, sg). A verb *dela-t'* (do) — *d'ela-ju* — *d'ela-jet* conjugates the same way, therefore it belongs to the same inflectional type. On the other hand, a verb *smotr'e-t'* (watch) — *smotr'-u* — *smotr'-it* conjugates differently and belongs to the other inflectional type. If we use these verbs to produce nouns with a derivational suffix *-imost'/-jemost'*, we will obtain *um'e-jemost'*, *d'ela-jemost'*, *smotr'-imost'*, and the stem changes here are the same as before.

The source code for our project is freely available [2].

To the best of our knowledge, this is the first work targeting the automatic induction of derivational morphology in East Slavic languages. In this paper, we propose our framework for deriving words in Russian and describe how to apply it to an arbitrary text corpus and extract derivational families. We illustrate this procedure on the example of Russian Wikipedia. Our results are promising, as the evaluation showed that our rule system covers a high percentage of the derived Russian words.

The structure of this paper is as follows. In section 2., we briefly overview the related work. In section 3., we outline our methodology. In section 4., we show how do we build a derivational morphology resource in our rule-based framework. In section 5., we describe our evaluation and its results. Finally, in section 6., we summarize our results and propose the directions of future work.

2. Related Work

One of the most prominent resources that contain information about derivational morphology in English, Dutch and German is CELEX (Baayen et al., 1995). However, it has a limited coverage and does not explicitly represent derivational relationships within families. In order to overcome these problems, Zeller et al. (2013) developed a rule-based system for German derivational morphology and obtained a high-coverage lexical resource. The same approach was later adapted to Croatian (Šnajder, 2014) resulting in DerivBase.Hr. An alternative resource for Croatian is CroDeriV, a morphological database of Croatian verbs (Šojat et al., 2014). Vidra et al. (2015) made a resource for Czech called DeriNet. Similarly, Lango et al. (2018) described the procedure for constructing word-formation networks for Spanish and Polish. Hathout and Namer (2014) proposed the French derivational morpho-semantic

[1] http://old.kpfu.ru/infres/slovar1/

[2] https://github.com/s231644/DerivBaseRu

network Démonette. Litta et al. (2016) made a resource for Latin. Talamo et al. (2016) published DerIvaTario, an annotated lexicon of Italian derivatives. Scharf (2017) implemented a derivational morphology rules of Sanskrit originally proposed by Panini. Haghdoost et al. (2019) built a morphological network for Persian on top of a morpheme-segmented lexicon.

DeriMo (Litta and Passarotti, 2017) and (Žabokrtskỳ et al., 2019) is a novel workshop that focuses on the recent advances of computational derivational morphology.

Let us now turn to available resources for Russian, the language in the focus of this paper.

Russian computational morphology in recent years has been mainly concentrated on inflectional processes and part of speech tagging. Pymorphy2 (Korobov, 2015) is an open-source morphological analyzer for Russian and Ukrainian. It allows to inflect and lemmatize words. MorphoRuEval-2017 (Sorokin et al., 2017) is an evaluation track for the automatic morphological analysis for Russian.

Another salient research area within computational morphology for Russian is an automatic morpheme segmentation. Arefyev et al. (2018) proposed to use a sequence to sequence neural network for that purpose. In contrast, Sorokin and Kravtsova (2018) applied a deep convolutional neural network. Both approaches achieved significant results. Although these works consider derivational morphology (they can detect derivational prefixes and suffixes), they do not target the induction of derivational families. The resource presented in this paper aims at filling this gap.

3. Framework

3.1. Russian Derivational Morphology

As said earlier, Russian has rich morphology. The morphological processes in Russian were extensively described in the grammar book by Shvedova et al. (1980). The section on derivational morphology is extremely detailed and contains about 300 pages. To simplify the problem, we decided to consider only productive types of derivation (that can produce new words in a modern language) and ignore rules for compounds and abbreviations.

The introduced rules are grouped first by a part of speech of a derived word, then by a word formation type, and, finally, by a part of speech of a source word.

In Shvedova et al. (1980), each derivational process is described in term of the following features:

- a subword constituent itself (e. g. a suffix *-k(a)/-ovk(a)/-jozhk(a)*);
- grammar characteristics of derived words (e. g. a gender of nouns);
- a general meaning of the type (e. g. 'a person doing some action', 'an abstract concept', 'a child of an animal');
- grammatical and semantic properties of the source words (e.g. inflectional types for nouns, adjectives and verbs);
- morphonological phenomena, e. g.:
  - *ú**kh**o (an ear)* → *ú**sh**ko* (a small ear) (velar—sibilant alternation),
  - *záya**ts** (a hare)* → *záya**ch**iy (hare's)* (ts—ch alternation),
  - *vý**lo**vit' (to fish out, perf.)* → *vy**lávl**ivat' (to fish out, imperf.)* (root vowel alternation, jotation),
  - *kósm**os** (space)* → *kosmícheskiy (space, adj.)* (stem deletion),
  - *dráma (a drama)* → *dram**at**ícheskiy (dramatic)* (stem epenthesis),
  - *Novosibírsk (Novosibrsk, noun)* → *Novosibír**[sk]**skiy (Novosibrsk, adj.), Odéssa (Odesa, noun)* → *Odés**[s]**skiy (Odesa, adj.)* (overlapping).
- productivity of the type and its subtypes (e. g. highly productive, productive, not productive);
- stylistic features and a scope of use of the type and its subtypes (e. g. mostly used in a scientific/ publicistic/ vernacular/ child language, etc.);
- additional information which falls out of the scope of this paper.

3.2. A Rule-based Derivation Model

Our approach is similar to the one that was used by Zeller et al, but we define more functions (because of a wide range of morphonological phenomena) and involve additional morphological information, namely, the inflectional types of nouns, adjectives and verbs.

Formally, for each *derivational pattern* p we define a rule as

$$r^p = (POS^p_s, POS^p_d, \{r^p_1, \ldots, r^p_{N^p}\}), \tag{1}$$

where POS^p_s and POS^p_d are the part of speech tags for source and derived words, respectively, N^p is a number of subrules, and r^p_i, $i = 1, \ldots, N^p$ are the subrules. Every subrule is a tuple

$$r^p_i = (TAGS^p_{i,s}, TAGS^p_{i,d}, t^p_i), \tag{2}$$

where $TAGS^p_{i,s}$ and $TAGS^p_{i,d}$ are additional tags (e.g. inflectional type, gender) for source and derived words, respectively, and t^p_i is a formal representation of changes that should be applied to the input word. We use a conjunction sign (&) to represent the sequational operations, and a disjunction sign (|) to represent the parallel operations. E.g. given a subrule representation

$$t = [o_1\&[o_2|o_3]|o_4]\&o_5,$$

the result of applying it on a word s will be

$$t(s) = o_5(o_2(o_1(s))) \cup o_5(o_3(o_1(s))) \cup o_5(o_4(s)).$$

We currently use a following set of operations (Table 1, some of the operations are not listed).

Each operation in a subrule has its own mode: *do* (default), *try* and *opt*. See Table 2

[3]This phenomenon in known as jotation.

Operation	Description
$delsfx(x)$	del. the suffix x
$addsfx(x)$	add the suffix x
$replsfx(x_1, x_2)$	repl. the suffix x_1 with x_2
$onlysfx(x)$	excl. if the suffix $\neq x$
$excsfx(x)$	excl. if the suffix $= x$
	similarly for prefixes
$plt()$	alt. (*g*, *zh*), (*k*, *ch*), (*kh*, *sh*)
$plt5()$	alt. (*ts*, *ch*)
$altcons()$	alt. (*m*, *ml*), ..., (*t*, *ch*), ...[3]
$invplt()$	inversed $plt()$
$invplt5()$	inversed $plt5()$
$invaltcons()$	inversed $altcons()$
$lsoft()$	alt. (*l*, *l'*)
$lhard()$	alt. (*l'*, *l*)
$delvowel()$	del. *o* or *e* on a position -2
$addvowel()$	add *o* or *e* on a position -2

Table 1: A partial list of implemented operations.

Apply o to s	do	try	opt
Possible	$o(s)$	$o(s)$	$o(s) \cup \{s\}$
Impossible	$\varnothing$	$\{s\}$	$\{s\}$

Table 2: Results of applying an operation o to a source word (string) s.

Thus, each rule takes an input as a source word, its POS tag and some other tags. The output is a set of words (with POS and other tags) that could be derived from the source word.

3.3. Graph Construction

Given a set of words $\mathcal{W}$ with known POS tags and additional information (gender, inflectional type, etc.). Let $\mathcal{R}$ be a set of implemented rules. Our goal is to obtain an oriented multigraph $\overrightarrow{\mathcal{G}} = (\mathcal{W}, \overrightarrow{\mathcal{E}})$ where $\overrightarrow{\mathcal{E}} \subset \mathcal{W} \times \mathcal{W} \times \mathcal{R}$. Every edge $e = (w_s, w_d, r) \in \overrightarrow{\mathcal{E}}$ can be interpreted as 'a word w_d could be derived from a word w_s with a rule r (or with a derivational pattern associated with it)'. A procedure of obtaining $\overrightarrow{\mathcal{G}}$ is shown in Algorithm 1.

Data: a set of words $\mathcal{W}$, a set of rules $\mathcal{R}$
Result: an oriented multigraph $\overrightarrow{G} = (\mathcal{W}, \overrightarrow{\mathcal{E}})$

```
E⃗ ⟵ ∅
for w_s ∈ W do
    for r ∈ R do
        D ⟵ r(w_s)
        for w_d ∈ D do
            if w_d ∈ W then
                E⃗ ⟵ E⃗ ∪ {(w_s, w_d, r)}
            end
        end
    end
end
```

Algorithm 1: Derivational multigraph induction

After the multigraph $\overrightarrow{\mathcal{G}}$ has been constructed, we can use it for searching words belonging to the same derivational families.

3.4. Induction of Derivational Families

Let us define a non-oriented graph $\mathcal{G} = (\mathcal{W}, \mathcal{E})$, where $\mathcal{E} \subset \mathcal{W} \times \mathcal{W}$ and $e = (w_s, w_d) \in \mathcal{E} \iff \exists r \in \mathcal{R} : (w_s, w_d, r) \in \overrightarrow{\mathcal{E}} \vee (w_d, w_s, r) \in \overrightarrow{\mathcal{E}}$.

3.4.1. Finding Connected Components

First, we decompose the graph $\mathcal{G}$ into n connected components $\mathcal{C}_1, \dots, \mathcal{C}_n$. It can be easily done, for instance, with a depth-first search (DFS).
However, decomposing the graph into connected components is not equivalent to induction of derivational families. For example, the nouns *vin-o* (wine) and *vin-a* (guilt) will be both connected with the adjective *vinniy*. Here we clearly see that the latter word belongs to two derivational families and to only one connected component.
The other issue was the structure of the connected components. Sometimes it is more convenient to represent derivational families as trees. But theoretically it is not required. E. g., a word *'nebystro'* can be derived in two ways:

1. *'bystriy'* (fast, adj.) $\rightarrow$ *'bystro'* (fast, adv.) $\rightarrow$ *'nebystro'* (not fast, adv.);
2. *'bystriy'* (fast, adj.) $\rightarrow$ *'nebystriy'* (not fast, adj.) $\rightarrow$ *'nebystro'* (not fast, adv.).

Even both-side derivations are possible: *Óm'* (the river) $\rightarrow$ *ómskiy* (related to Om', adj.) $\rightarrow$ *Ómsk* (a city) $\rightarrow$ *ómskiy* (related to Omsk, adj.).
For this reasons, we publish three versions of our dataset:

- only connected components extracted;
- + derivational families induced (one graph for one root);
- + derivational families are tree-structured.

3.4.2. Extracting Derivational Families

Let $\mathcal{C}$ be one of the connected components and $\overrightarrow{\mathcal{C}}$ be its directed variant. Our goal is to find a set of vertices that we will call roots of derivational families.
Our algorithm consists of 4 steps.

1. Decompose a graph into the strongly connected components (SCCs) $\mathcal{S}_1, \dots, \mathcal{S}_m$.
2. Replace $\mathcal{S}_i, i = 1, \dots, m$ with one new vertex s_i and obtain a directed acyclic graph.
3. Find all roots—the vertices with zero indegree.
4. If s_i is one of the roots, replace it with any word $z_i \in \mathcal{S}_i$.

Now that we have roots, we can run DFS from each one, and all reachable vertices will come to the corresponding derivational families.
To obtain derivational trees, for each vertex v in a particular derivational family we simply memorize its parent, a vertex that was visited on a previous step of DFS.

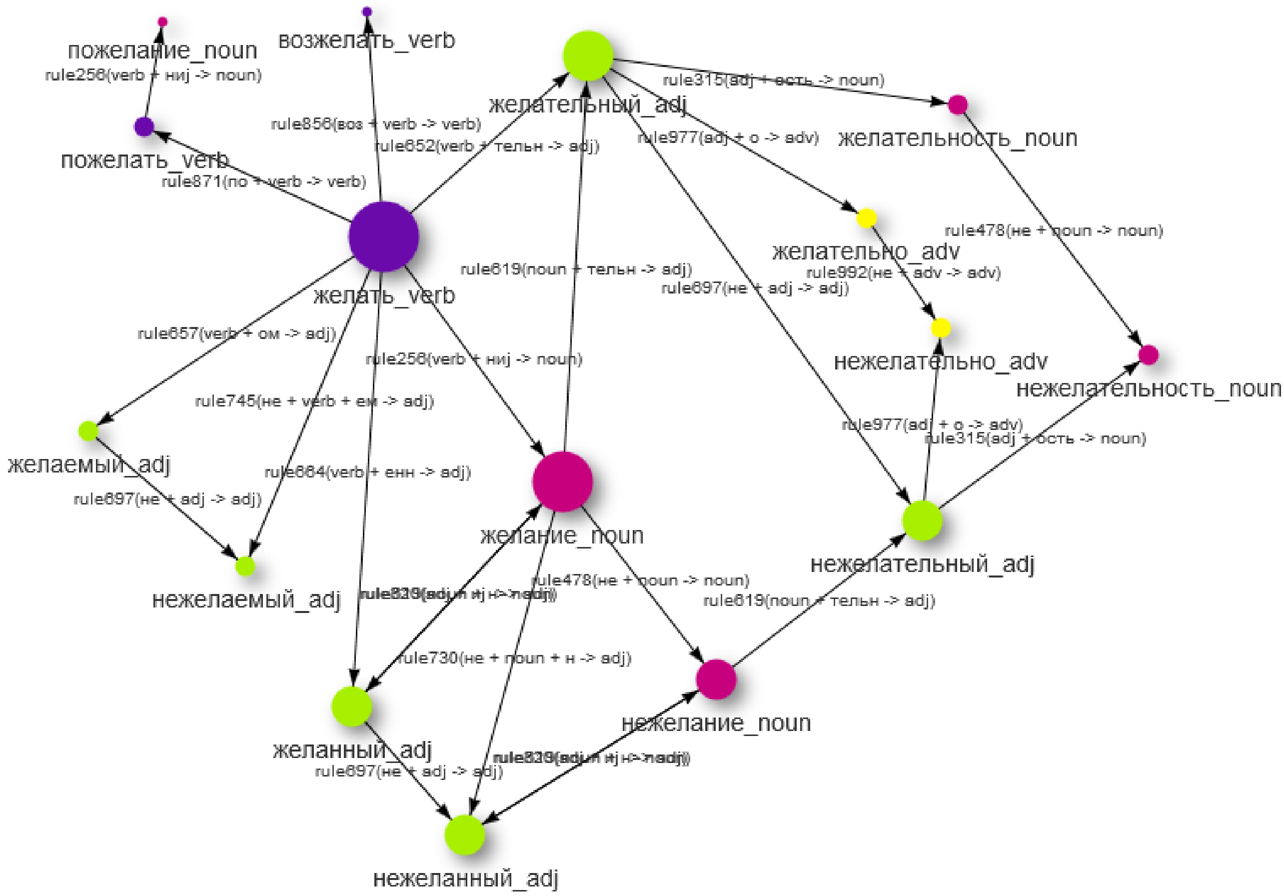

Figure 1: Visualization of the connected component for a word *zhelat'* (to wish).

4. Building the Resource

4.1. Preparing the Data

To construct the graph $\vec{\mathcal{G}}$, we first need to collect a vocabulary $\mathcal{W}$.

In order to do that, we took a Russian Wikipedia dump on May 2019[4] and removed all non-textual information (XML tags, hyperlinks, etc.) with WikiExtractor[5]. After that, we used a WebVectors[6] script that makes tokenization, lemmatization and POS-tagging with UDPipe. It also allows to extract collocations, e.g. *New::York_PROPN*.

We considered only five POS tags: *PROPN* (proper nouns; later replaced with *NOUN* labels), *NOUN* (nouns), *ADJ* (adjectives), *VERB* (verbs) and *ADV* (adverbs). Also, we removed collocations and not-cyrillic written words (e.g. *5-j_ADJ* (5th), *Google_PROPN*). We also merged proper nouns with ordinary nouns. The final number of words is shown in Table 3.

To correctly determine the inflectional types, we implemented a special set of rules for nouns, adjectives and verbs. While for the lemmatized nouns and adjectives inflectional types can be determined easily, verbs require an infinitive, a 1st person, and a 3rd person forms. For verbs inflection we use the Pymorphy2 (Korobov, 2015) library.

[4]https://dumps.wikimedia.org/ruwiki/20190501/

[5]https://github.com/attardi/wikiextractor

[6]https://github.com/akutuzov/webvectors

PROPN	NOUN	ADJ	VERB	ADV
23085	128806	82533	40985	12532

Table 3: Amounts of words of the considered parts of speech.

4.2. Implementation of Rules

We implemented our rules using The Russian Grammar book (Shvedova, 1980). The rules are stored in a JSON format and include POS tags for a source and derived words, and subrules with the additional tags of a source word. Subrules were parsed with Lark[7] and then represented as abstract syntax trees. The rules are splitted on several files corresponding to POS tags of derived words. Totally we obtained about 600 rules.

4.3. Statistics

After applying the describe earlier procedures, we obtained 613k words and 183k connections between them. After filtering the words that occur in the training corpora less than 5 times, we obtained 41k 'active' nodes, 46k arcs and 6k connected components. Without threshold, we would obtain 275k nodes, 159k arcs and 15k connected components. Distribution of the rules can be seen at Figure 2.

[7]https://github.com/lark-parser/lark

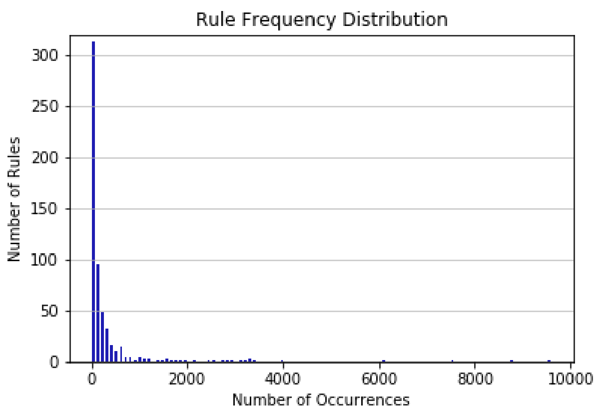

Figure 2: Historgam for rule frequency distribution.

сомнева'ться

сомне'-ниj-е [сомнение]
не-сомне'н-н(ый)
несомне'нн-о
несомне'нн-ость
сомн-и'/тельн(ый)
сомни'тельн-о
сомни'тельн-ость
не-сомни'тельн(ый)
несомни'тельн-о
несомни'тельн-ость
по-сомнева'ться
у-сомн-и'-ться
по-усомни'ться

Figure 3: Derivational tree for word *somnevat'sya* (to doubt).

4.4. Visualization

We visualize graphs using visJS2jupyter (Rosenthal et al., 2018) and save them as HTML pages. All graphs can be found in the main project repository. Each node has one of four colours corresponding to their POS tags. Nodes are labelled with words, and edges are labelled with rules. The size of the node is proportional to the number of its input and output degrees.

5. Evaluation

5.1. Word Formation Dictionary of Russian

For evaluation we used Word Formation Dictionary of Russian[8] that contains more than 1500 derivational families that are structured as trees. See Figure 3 for an example.
From these trees we extracted 49611 pairs (w_s, w_d) of the source and derived words ignoring such pairs where derivation was made with compounding (compounded words can be easily found because they have a special label). After that, we checked whether the set of derived words for w_s contains w_d. We obtained **0.8134** recall. If we search both (w_s, w_d) and (w_d, w_s) pairs, the recall becomes **0.8209**.
Further analysis showed that there are several main error sources.

1. Not considered rules (including abbreviation and compounding with Latin or Greek roots that were not marked directly).
2. Different conventions about derivational processes in the dictionary and in the Russian Grammar.
3. Complex root changes and other phenomena (not fully implemented).
4. Rare and irregular derived words (not fully implemented).
5. Rare and unproductive derivational types.
6. Misspellings in the test set.

5.2. Wiktionary (Adverbs)

Our second evaluation was made for adverbs from Wiktionary. We took a list of adverbs, excluded obscene, vernacular, archaic non-motivated words, and words with unclear origin. Finally we got 6747 words and manually reconstructed source words for them. After that, we did the same as for the Word Formation Dictionary. Our recall was **0.9214**. We compared our system with the baseline Tikhonov morphemic dictionary[9]. We searched whether the target word was in this dictionary, and the result was **0.8472** recall. So our system significantly outperformed the dictionary.
Error analysis showed that most of the negative examples are produced by unproductive or irregular derivational types that were not implemented in our system.

6. Conclusion

In this work we presented our framework of designing a rule-based system for modeling Russian derivational morphology. The source code for our project is freely available online. We applied this framework to Russian Wikipedia and visualized obtained connected components. We named this resource DerivBase.Ru. We evaluated our system on two test sets and received high recall.
We believe that our work will be useful for linguists, researchers and engineers.
We are planning to further develop our system by updating the rules and manually adding the unproductive derivational pairs.
Having large dataset of source—derived pairs with corresponding rule IDs, it is possible to train a neural network that would be more robust than a rule-based system. Also we aim to train a neural network to produce the inverted transformations. The architecture of such a network can be adapted, for instance, from (Cotterell et al., 2017). We also want to apply our resource to other downstream NLP tasks, such as language modeling, dependency parsing and machine translation.

7. Acknowledgments

This research was financially supported by the Russian Foundation for Basic Research (RFBR) and the Government of Novosibirsk Region (Project No. 18-41-543015).

[8] http://old.kpfu.ru/infres/slovar1/

[9] http://www.slovorod.ru/der-tikhonov/index.html

POSs	POSd	Derivation	Number of rules
VERB	NOUN	SFX	62
ADJ	NOUN	SFX	41
NOUN	NOUN	SFX	106
VERB	NOUN	0SFX	3
ADJ	NOUN	0SFX	6
NOUN	NOUN	GEN	2
NOUN	NOUN	PFX	31
ADJ	NOUN	PFX+SFX	3
VERB	NOUN	PFX+SFX	2
NOUN	NOUN	PFX+SFX	30
VERB	NOUN	PFX+0SFX	6
ADJ	NOUN	GEN	4
NOUN	ADJ	SFX	32
VERB	ADJ	SFX	23
ADJ	ADJ	SFX	9
ADJ	ADJ	PFX	35
NOUN	ADJ	PFX+SFX	24
VERB	ADJ	PFX+SFX	5
NOUN	ADJ	PFX+0SFX	1
NOUN	VERB	SFX	6
ADJ	VERB	SFX	6
VERB	VERB	SFX	3
VERB	VERB	PFX	29
NOUN	VERB	PFX+SFX	21
ADJ	VERB	PFX+SFX	20
VERB	VERB	PFX+SFX	31
VERB	VERB	PTFX	1
NOUN	VERB	SFX+PTFX	1
ADJ	VERB	SFX+PTFX	1
VERB	VERB	PFX+PTFX	16
NOUN	VERB	PFX+SFX+PTFX	2
ADJ	ADV	SFX	3
NOUN	ADV	SFX	1
VERB	ADV	SFX	2
ADV	ADV	SFX	4
ADV	ADV	PFX	1
ADJ	ADV	PFX+SFX	11
NOUN	ADV	PFX+0SFX	2
VERB	ADV	PFX+SFX	2

Table 4: Number of implemented rules grouped by POS of the source and target words, and type of derivation (prefixation, suffixation, zero suffixation, postfixation, gender modification (with no derivational suffixes added))

8. Bibliographical References

Arefyev, N., Gratsianova, T., and Popov, K. (2018). Morphological segmentation with sequence to sequence neural network.

Baayen, R. H., Piepenbrock, R., and Gulikers, L. (1995). The celex lexical database (release 2). *Distributed by the Linguistic Data Consortium, University of Pennsylvania*.

Cotterell, R., Vylomova, E., Khayrallah, H., Kirov, C., and Yarowsky, D. (2017). Paradigm completion for derivational morphology. In *EMNLP*.

Haghdoost, H., Ansari, E., Žabokrtskỳ, Z., and Nikravesh, M. (2019). Building a morphological network for persian on top of a morpheme-segmented lexicon. In *Proceedings of the Second International Workshop on Resources and Tools for Derivational Morphology*, pages 91–100.

Hathout, N. and Namer, F. (2014). Démonette, a french derivational morpho-semantic network. *LiLT (Linguistic Issues in Language Technology)*, 11.

Korobov, M. (2015). Morphological analyzer and generator for russian and ukrainian languages. In Mikhail Yu. Khachay, et al., editors, *Analysis of Images, Social Networks and Texts*, volume 542 of *Communications in Computer and Information Science*, pages 320–332. Springer International Publishing.

Lango, M., Sevcikova, M., and Žabokrtskỳ, Z. (2018). Semi-automatic construction of word-formation networks (for polish and spanish). In *Proceedings of the*

Eleventh International Conference on Language Resources and Evaluation (LREC-2018).

Litta, E. M. and Passarotti, M. C. (2017). Proceedings of the workshop on resources and tools for derivational morphology (derimo). 5-6 october 2017, milano, italy.

Litta, E., Passarotti, M., and Culy, C. (2016). Formatio formosa est. building a word formation lexicon for latin. In *Proceedings of the third italian conference on computational linguistics (clic–it 2016)*, pages 185–189.

Rosenthal, S. B., Len, J., Webster, M., Gary, A., Birmingham, A., and Fisch, K. M. (2018). Interactive network visualization in jupyter notebooks: visjs2jupyter. *Bioinformatics*, 34(1):126–128.

Scharf, P. M. (2017). A computational implementation of pan. ini's derivational morphology of sanskrit. *on Resources and Tools for Derivational Morphology (DeriMo)*, page 93.

Shvedova, N. (1980). *Russkaja grammatika*. Number t. 1 in Russkaja grammatika. Izd-vo Nauka.

Šnajder, J. (2014). Derivbase.hr: A high-coverage derivational morphology resource for croatian. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland, may. European Language Resources Association (ELRA).

Šojat, K., Srebačić, M., Pavelić, T., and Tadić, M. (2014). Croderiv: a new resource for processing croatian morphology. *Proceedings of the Language Resources and Evaluation-LREC*, 14:3366–3370.

Sorokin, A. and Kravtsova, A. (2018). Deep convolutional networks for supervised morpheme segmentation of russian language. In Dmitry Ustalov, et al., editors, *Artificial Intelligence and Natural Language*, pages 3–10, Cham. Springer International Publishing.

Sorokin, A., Shavrina, T., Lyashevskaya, O., Bocharov, B., Alexeeva, S., Droganova, K., Fenogenova, A., and Granovsky, D. (2017). Morphorueval-2017: an evaluation track for the automatic morphological analysis methods for russian.

Talamo, L., Celata, C., and Bertinetto, P. M. (2016). Derivatario: An annotated lexicon of italian derivatives. *Word Structure*, 9(1):72–102.

Tikhonov, A. (1985). *Slovoobrazovatel'ny Slovar Russkogo Jazyka. (Word-Formational Dictionary of Russian Language)*. Number t. 1 in Slovoobrazovatel'ny Slovar Russkogo Jazyka. (Word-Formational Dictionary of Russian Language). M.: Russkij Jazyk.

Tikhonov, A. (2017). *Novy Slovoobrazovatel'ny Slovar Russkogo Jazyka Dlya Vseh, Kto Hochet Byt' Gramotnym*. LitRes.

Vidra, J., Žabokrtský, Z., Ševčíková, M., and Straka, M. (2015). DeriNet 1.0. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Žabokrtskỳ, Z., Ševčíková, M., Litta, E., and Passarotti, M. (2019). Proceedings of the second international workshop on resources and tools for derivational morphology (derimo 2019). 19-20 september 2019, prague, czechia.

Zaliznyak, A. (1980). *Grammaticheskiy Slovar' Russkoro Jazyka: Slovoizmeneniye: okolo 100 000 slov*. Russkiy Jazyk.

Zeller, B., Šnajder, J., and Padó, S. (2013). DErivBase: Inducing and evaluating a derivational morphology resource for German. In *Proceedings of ACL 2013*, pages 1201–1211, Sofia, Bulgaria.