Automatic detection of morphological paradigms using corpora information

Sorokin A. A. (alexey.sorokin@list.ru)[1,2],
Khomchenkova I. A. (irina.khomchenkova@yandex.ru)[1]

[1] Lomonosov Moscow State University, Moscow, Russia,
[2] Moscow Institute of Physics and Technology, Dolgopudnyj, Russia
Annotation: This paper deals with automatic induction and prediction of morphological paradigms for Russian. We apply a method of longest common subsequence to extract abstract paradigms from inflectional tables. Then we experiment with automatic detection of paradigms using a linear classifier with lexeme suffixes and prefixes as features. We show that Russian noun paradigms could be automatically detected with 77% accuracy per paradigm and 93% accuracy per word form, for Russian verbs per-paradigm accuracy reaches 76% and per-form accuracy is 89%. Usage of corpora information and character ngrams allows to improve these results up to 82% and 95% for nouns and 86% and 95% for verbs.
Keywords: abstract paradigm, paradigm induction, longest common subsequence, automatic paradigm detection, corpora-based paradigm detection

Автоматическое определение морфологических парадигм с использованием корпусной информации

Сорокин А. А. (alexey.sorokin@list.ru)[1,2],
Хомченкова И. А. (irina.khomchenkova@yandex.ru)[1]

[1] Московский государственный университет им. М. В. Ломоносова, Москва, Россия
[2] Московский физико-технический институт, Долгопрудный, Россия
Annotation: Данная работа посвящена автоматическому определению и классификации морфологических парадигм для русского языка. Абстрактные морфологические парадигмы выделяются с помощью метода наибольшей общей подпоследовательности. Основная часть работа посвящена проблеме вычисления полной парадигмы для неизвестной лексемы, для чего применяется линейная классификация. В качестве признаков для классификации используются префиксы и суффиксы данной лексемы. Мы показываем, что абстрактная парадигма может быть определена с точностью 77% для существительных и 76% для глаголов, в то время как точность по словоформам достигает 93 и 89%. В работе вводится новый

алгоритм автоматического определения морфологической парадигмы, использующий корпусную информацию. Он позволяет достичь качества в 82% для именных и 86% для глагольных парадигм, в то время как точность по словоформам в обоих случаях становится равной 95%.
Keywords: морфологическая парадигма, абстрактная парадигма, автоматическое определение парадигм, автоматическая классификация парадигм, корпусной метод определения парадигм

1 Introduction

The automatic induction and learning of morphological paradigms is very popular in the last years. State-of-the-art works include [Ahlberg et al., 2015] and [Nicolai et al., 2015], but several other papers are woth mentioning ([Ahlberg et al., 2014], [Durrett, DeNero, 2013]). This task has various applications, e.g. synthesis of surface word forms in machine translation and automatic extension of morphological resources, such as `wiktionary.org`. The methods developed for paradigm learning can also be used in automatic morphological analysis, e.g. for POS-tagging or lemmatization.

The automatic induction of morphological paradigms has a long history in the Russian linguistic tradition. The seminal work of A. A. Zaliznyak "Russkoe imennoe slovoizmenenie" [Zaliznyak, 2002] solves exactly this problem: how the complete description of morphological inflection could be recovered from empirical data. If we reconsider the algorithm of Zaliznyak from the computational point of view and omit the technical details specific to Russian phonology, it is essentially based on the method of longest common subsequence (LCS): the invariant part of inflected forms of the same lexeme is exactly their LCS. The method of LCS for automatic induction of morphological paradigms was reintroduced in works of Ahlberg, Hulden et al. ([Ahlberg et al., 2014], [Ahlberg et al., 2015]). However, for the purposes of computational lingustics, automatic induction of morphological paradigms from inflected tables is only the preliminary step. A more important question is how to detect the paradigm label and hence the complete inflectional table using only the base form of the lexeme. This problem is solved by machine learning techniques, using substrings of the source lexeme (e.g., its prefixes or suffixes) as features for the classifier.

There are practically no works on automatic detection of morphological paradigms for Russian: [Ahlberg et al., 2015] contains some results for noun inflection but the quality of the source data is too low to consider them significant. We reimplement the method of Hulden for paradigm induction

with several technical modifications and use a linear classifier to derive these paradigms automatically from the lexeme. Our algorithm is able to recover complete morphological paradigm both for Russian nouns and verbs with accuracy of 77% for paradigms and 93 and 88% for word forms respectively. We also demonstrate that the usage of corpora information improves the percentage of correctly predicted paradigms up to 81% for nouns and 83% for verbs.

2 Abstract paradigms

For the compressed representation of morphological inflection we use the notion of an abstract paradigm, introduced in [Ahlberg et al., 2014]. From the mathematical point of view, a paradigm is a tuple of functions $F = \langle f_1, ..., f_n \rangle$ taking the same variables $x_1, \ldots, x_r \in \Sigma^+$, where $f_i(x_1, ..., x_r)$ operates from $(\Sigma^+)^r$ to Σ^+ ([Ahlberg et al., 2014], see also [Zaliznyak, 2002]). Here Σ is the finite alphabet and Σ^+ denotes the set of all words over this alphabet. Each of the functions f_i corresponds to some grammatical meaning c_i, the functions in set F are arranged according to a fixed order $c_1, \ldots, c_n$ of possible grammatical meanings. Literally speaking, a paradigm is a mapping from variables to strings. We use the term "abstract paradigm" to represent morphological paradigms formally. An abstract paradigm is a tuple of strings containing variables $x_1, x_2, ...x_n$ (the variables are the same for all strings and have the same order elsewhere) and constant fragments, which are the same for all lexemes satisfying the given paradigm. These constant fragments vary between the forms of the same lexeme. On the contrary, the variables have the same value for all inflected forms but differ from lexeme to lexeme.

Let us explain these formal terms on a short example. Consider the inflectional tables of two Russian nouns *кусок* and *песок*. The paradigm function F is the same for both of them; in the first case it takes the variables $x_1 =$ *кус* and $x_2 =$ *к*, in the second one — $x_1 =$ *пес*, $x_2 =$ *к*.

Given the variable values, an abstract paradigm unambigiously determines the complete inflectional table. When a pattern and a word form are known, usually there is only one way to fit the pattern to the word: for example, the word *мешок* and the pattern x_1+о+x_2 yield a single combination of variable values $x_1 =$ *меш*, $x_2 =$ *к*. Nevertheless, applying the same pattern to the word *носок* results in two variants $x_1 =$ *н*, $x_2 =$ *сок* and $x_1 =$ *нос*, $x_2 =$ *к*. If we take into account several possible patterns, the number of decompositions can grow up dramatically. However, the variables are extracted not from a single word form, but from all the paradigm elements simultaneously, which

Grammeme	Pattern	$F(кус,к)$	$F(пес,к)$
Nom.Sg.	$x_1 + \mathbf{о} + x_2$	кусок	песок
Nom.Pl.	$x_1 + x_2 + \mathbf{и}$	куски	пески
Gen.Sg.	$x_1 + x_2 + \mathbf{а}$	куска	песка
Gen.Pl	$x_1 + x_2 + \mathbf{ов}$	кусков	песков
Dat.Sg.	$x_1 + x_2 + \mathbf{у}$	куску	песку
Dat.Pl.	$x_1 + x_2 + \mathbf{ам}$	кускам	пескам
Acc.Sg.	$x_1 + \mathbf{о} + x_2$	кусок	песок
Acc.Pl.	$x_1 + x_2 + \mathbf{и}$	куски	пески
Instr.Sg.	$x_1 + x_2 + \mathbf{ом}$	куском	песком
Instr.Pl.	$x_1 + x_2 + \mathbf{ами}$	кусками	песками
Pr.Sg.	$x_1 + x_2 + \mathbf{е}$	куске	песке
Pr.Pl	$x_1 + x_2 + \mathbf{ах}$	кусках	песках

Table 1: Abstract paradigm: an example

restricts the set of possible combinations.

2.1 Longest common subsequence

Consider again the abstract representation of morphological paradigms. If we substitute strings of letters for the variables, these strings form a common subsequence of all generated words. In order to capture as much common material as possible, that subsequence should be the longest one. Therefore, the problem of paradigm detection has been reduced to the task of finding the longest common subsequence. We are not going to discuss the linguistic relevance of this approach and use it only as an empirical procedure. However, several important questions emerge:

1. How to calculate the longest common subsequence algorithmically?
2. What subsequence to select when several subsequences have the same length?
3. How to extract variable values when the LCS is known?

For the first task we use finite automata. It is straightforward to construct an automaton recognizing all the common subsequences of given strings and then extract the longest word this automaton accepts (we omit algorithmical details). Although, this automaton could be nondetermenistic and an equivalent deterministic state automaton may have much larger number of

states (up to 2^n where n is the number of states of initial nondeterministic automaton). To prevent this exponential growth we bound the length of gaps between the consequent letters of the subsequence, as well as the gap before the first letter of the subsequence. This limitation is also justified from the linguistic point of view: consider two verb forms *разместиться* and *размещусь*, their LCS *размес* has length 6. However, *с* in the LCS is an artifact of the method, not an element of common stem. Besides, alterations like *ст/щ* are among the phenomena which are difficult to capture by LCS algorithm.

The construction of finite automata recognizing all common subsequences for the words *моток* and *окот* is illustrated below. The edges contain not only the symbols, but also the positions of these symbols in the words. This trick allows to simplify the extraction of an abstract paradigm from the LCS.

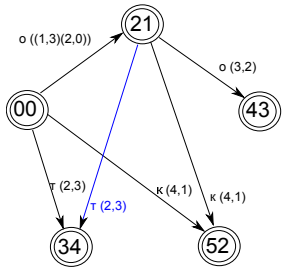

Figure 1: DSA for common subsequences of the word *моток* and *окот*

In the example above there are 3 longest common subsequences: *оо*, *ок*, *от*. Possible variants of their positioning are shown in the table below.

LCS	LCS positioning variants
О-О	М**О**Т**О**К, **О**К**О**Т
О-Т	М**ОТ**ОК, **О**КО**Т**
О-Т	М**ОТ**ОК, ОК**ОТ**
О-К	М**О**ТО**К**, **ОК**ОТ
О-К	МОТ**ОК**, **ОК**ОТ

Table 2: LCS for the words *моток* and *окот*

Already in this artificial example there are multiple variants for LCS positioning. The same problem emerges in practice: consider a partial inflectional table of the word *песок*. There are two candidates for the LCS: *пес-о* and *пес-к* both of length 4.

песок	Nom.Sg.
песков	Gen.Pl.
песком	Instr.Sg.

песок	Nom.Sg.
песков	Gen.Pl.
песком	Instr.Sg.

Table 3: Ambigious LCS positioning: an example

We use two euristics for disambiguation: the first selects the variant with the minimal number of variables (variables are the maximal contiguous parts of the LCS). However, this euristic does not give us a solution here: both subsequences consist of two variables. Then we apply our second euristic: choose the variant with the least total length of gaps. Then the variant **песок-песк**ов-**песк**ом is preferred, since it leads to a single gap of length 1 while its counterpart generates two such gaps (of total length 2).

3 Automatic detection of paradigms

In the previous section we have discussed the algorithm for morphological paradigms induction. However, it is not a central problem of the paper; we are mainly interested in the automatic detection of such paradigms for unknown words. We consider the following task: given an unknown word of a known part-of-speech (say, a noun *арка*), determine its complete inflectional table. The algorithm selects one of many potential variants, several of which are listed in Table 4.

Paradigm	Variables
1#1+ы#1+а#1+ов#1+у#1+ам#1#1+ы#1+ом#1+ами#1+е#1+ах	1=арка
1+а#1+а#1+ы#1#1+е#1+ам#1+у#1+ы#1+ой#1+ами#1+е#1+ах	1 = арк
1#1+ы#1+а#1+ов#1+у#1+ам#1#1+а#1+ов#1+ами#1+е#1+ах	1 = арка
1+2+а#1+2+и#1+2+и#1+о+2#1+2+е#1+2+ам#1+2+у#1+2+и #1+2+ой#1+2+ами#1+2+е#1+2+ах	1 = ар, 2 = к

Table 4: Multiple possible paradigms for the word *арка*

We may attempt to recover a correct paradigm using deterministic rules such as "when a noun ends with *а* then this *а* is a flection, not a part of a stem" (counterexample: *баккара*) and if such word ends with "*Ска*" for some consonant *С* then *о* is inserted between *С* and к in genitive plural (counterexample: *ласка*). However, all such rules have counterexamples and their manual design is a very labour-intensive task. Therefore we have decided to learn inflectional patterns automatically applying algorithms of

machine learning. We use as features all the suffixes[1] whose length does not exceed the given maximum (say, 5). The suffixes are encoded as binary indicators; for example, the word *учитель* is described by a binary vector with five nonzero elements, corresponding to suffixes *-ь, -ль, -ель* etc. (see Table 5 below). The absense of a suffix in the training set is encoded by a special placeholder, in this case longer suffixes are not taken into account since they were not observed in the training set either. For example, is the suffix *-ль* was preceded only by *е* in the training set, than both the words *моральь* and *фасольь* are encoded by vector containing three ones for suffixes *-ь, -ль* and *!ль* where ! denotes an unobserved letter.

	$а	$к	$ка	$ла	$ик	$рка	...
арка	1	0	1	0	0	1	...
школа	1	0	0	1	0	0	...
блик	0	1	0	0	1	0	...
...	...	...	...	...	...	...	...

Table 5: Feature encoding scheme

Since prefixes carry no information about noun morphology, we do not use them as features for noun paradigm prediction. In the case of verbs, conversely, they can be used to determine verb aspect. If d is the maximal length of suffixes used as features, then the number of possible features grows roughly exponentially with d and may reach 20000 for $d = 5$. To reduce training time and remove noisy features we retain only a fixed percentage of the most unambigious features. As the measure of ambiguity for the feature f_j we take $\max_i P(c(L) = c_i | f_j(L) = 1)$ — the probability of the most frequent class provided f_j is present. We also remove the features which appear less than 3 times in the training set.

4 Evaluation of paradigm classifier

We evaluated our approach on Russian verbs and nouns. For both tasks we took 5000 most frequent words of the corresponding part of speech from the dictionary of Lyashevskaya and Sharoff ([Lyashevskaya,Sharoff, 2009]). We automatically downloaded complete inflectional tables from the Wiktionary

[1] we use the term "suffix" ("prefix") for an arbitrary substring in the end (in the beginning) without any regard to morphology

(ru.wiktionary.org). For nouns the tables contained at most 12 items for 6 cases and 2 numbers (several cells in the paradigm could be empty, e.g. for pluralia tantum). Sometimes the cell contained two values (for example, Instr.Sg. of first declension nouns), in this case we always chose only the first form. We extracted 239 abstract paradigms for noun declension, 69 of them contain more than 5 examples and 108 – only a single example. 10 most frequent paradigms are listed in 11 of the Appendix.
In the case of verbs typical Wikitionary form for imperfect aspect contains 21 simple forms (https://ru.wiktionary.org/wiki/%D0%B6%D0%B5%D0%BB%D0%B0%D1%82%D1%8C) including infinitive and omitting composite future form and empty cells. For paradigm induction we used only 13 of them: 6 present forms, 4 past, 2 forms of the imperative and the basic infinitive form. Even in such restricted form verb conjugation demonstrate more irregularities then noun declension, so the sample of 5000 verbs contains 305 paradigms with 120 of them having 5 or more representatives and 92 — a single representative. 10 most frequent paradigms are shown in Table 12. We bound maximal gap length by 2, therefore the algorithm does not recoginze c as part of the LCS in the examples like *играться/играешься/играйтесь*.
In our experiments we randomly separated the sample on 2 equal halfs, using one for testing and the other one for training. The results were averaged for 5 random splits. In the case of nouns we did not use prefixes as features and bound suffix length d by $3, 5$ or 7. The percentage p of selected features was $0.10, 0.25$ or 0.5. In the case of verbs we calculated the suffix length without the reflexive affixes *-ся* and *-сь*. We also used the prefix features with the maximal length of 2 for verb conjugation. To predict paradigm labels we used the logistic regression classifier from sklearn package [Pedregosa et al., 2011], which itself uses the LIBLINEAR library [Fan et al., 2008]. The results are presented in Table 6 and Table 7. We report both per-paradigm (the percentage of correctly predicted abstract paradigms) and per-form (the fraction of correct word forms) accuracy.

	0.1	0.25	0.5
3	77.19 93.47	77.26 93.47	77.25 93.47
5	77.38 93.50	77.32 93.48	77.32 93.48
7	77.44 93.45	77.35 93.43	77.35 93.43

Table 6: Prediction accuracy for noun paradigms classification

Since the result of nouns is practically independent from the classifier pa-

rameters, we fix $p = 0.1$ and $d = 5$ in future experiments. We use the same setting for the verbs task, however, in this case the impact of feature length is more significant.

	0.1	0.25	0.5
3	51.41 79.96	51.41 79.96	51.41 79.94
5	76.30 88.83	76.09 88.62	75.94 88.62
7	77.06 88.36	78.01 89.35	77.96 89.38

Table 7: Prediction accuracy for verb paradigms classification

We also study how the prediction quality changes with the size of the training set. When there is little training data available, a lemma may not fit to all inflection patterns observed in training phase (say, a verb ends with *-ти* and all the infinitives in the training set ended with *-ть*, *-ться* or *-чь*). In such cases we allow the system consult a complete list of paradigms, no matter whether they were observed in training. The dependence between training data size and system performance is shown in Table 8.

Task	Training data fraction					
	0.1	0.25	0.5	0.6	0.7	0.8
Nouns	71.76 91.15	75.05 92.32	77.38 93.50	77.95 93.70	77.88 93.77	77.40 93.84
Verbs	65.50 83.83	71.50 86.27	76.30 88.83	77.49 89.36	77.60 89.41	77.56 89.50

Table 8: Train data percentage and performance quality

4.1 Analysis of results

It is uninformative to compare results for different languages and even for different datasets. As we know, the only experiment on paradigm detection for Russian nouns was conducted by Ahlberg et al. in [Ahlberg et al., 2015], showing per-table accuracy of 66% and per-form accuracy of 89%. However, they used data collected from Freeling ([Padro, Stanilovsky, 2012]), which is of much lower quality than ours. They also used 5-fold cross-validation for performance evaluation, which means that 80% was left for training instead of only 50% in our experiment. However, the results for other languages,

such as Catalan, French or Italian, reported in [Ahlberg et al., 2015] are much higher with per-table accuracy of over 90%. We claim that corpus-free methods are incapable of reaching comparable accuracy on Russian data due to objective linguistic factors.

The are two main sources of errors in the case of noun paradigm prediction: the first is animacy/unanimacy affecting the forms of accusative, the second is -а/-ы in the form of Nom.Pl. In both cases the correct category does not depend on the surface form (consider *волчонок* vs *бочонок* or *голос* vs *колос*). The system also fails to discriminate between masculine and feminine nouns ending with ь (*мозоль* vs *король*). It is obvious that these ambiguities cannot be resolved without corpus statistics. We discuss this question in details in the next section.

For verbs the problem is more subtle. Often the mistake happens for the forms of imperative mood, for example, **тревожи* is predicted instead of *тревожь* or **похити* for *похить*. In such cases the forms or indicative mood are usually correct. Another common source of mistakes are *е/ё* in verb flections (compare *хлопнуть* and *толкнуть*). In this case the flection depends on the stress position in the infinitive form, however, we removed the stress signs in our data since they are marked inconsistently in Wiktionary itself. Such mistakes affect only several forms (imperative or third person presens). Errors of the second type touch practically all forms of the paradigm. It often happens for the verbs ending on -ать (*венчать* vs *кричать*). The system also fails in the case of phonetic alterations (*унизить/унижу*), especially when they happen inside the stem (*звать/зову* or *слать/шлю*).

Summarizing, the spectrum of possible errors for Russian verb paradigm prediction is wider than for Russian nouns, which explains lower per-form quality in the verb prediction task. However, in both cases more training data does not help, as shown in Table 8. We consider the sources of additional information in the next section.

5 Corpus-based methods of paradigm predictions

In this section we experiment with other features which might be helpful for automatic paradigm detection. In the verb paradigm task incorrectly predicted forms sometimes violate the rules of Russian phonology like in **осуществься* or **исчежь* for *исчезни*. These incorrect forms might be rejected if we extend the model by phonological features. This idea is realized as following:

First, we train a character ngram model on the training data. Then we

augment the algorithm with second classifier on the top of the first. It takes as features logarithmic probabilities predicted by the classifier on the first level as well as the scores of the language model. If the basic classifier has predicted c_i as paradigm label for the lemma L, we generate all the forms $w_{i,1}, \ldots, w_{i,m}$ of this lexeme according to the paradigm; then we take as language model score the averaged sum $s(L, c_i) = \frac{\sum_{j=1}^{m} -\log P_{lm}(w_{i,j})}{m}$ where $P_{lm}(w_{i,j})$ is the probability of wordform $w_{i,j}$ according to character ngram model. We test two ways of accomodating the language model log-scores: in the first case we use them as features of the linear classifier. In the second variant we used language model scores only for filtering, discarding a paradigm c_i if its score $s(L, c_i)$ is greater than $s_0 + \alpha$ where s_0 is the lowest value among $s(L, c_i)$ and α is some redefined constant. We used 5-gram language models trained on the set of word forms from the training data and smoothed the model counts using Witten-Bell smoothing ([Chen,Goodman, 1996]. The results for Nouns and Verbs tasks are presented in Table 9, we used $p = 0.1$ and $d = 5$ for feature fraction and suffix length in all trials, the percentage of training data was again 0.5.

Task	No character scores	Character scores as features	Character scores as filters
Nouns	77.38 93.50	77.42 93.50	77.36 93.42
Verbs	76.30 88.86	80.37 90.92	77.01 89.35

Table 9: Using character model for paradigm prediction

We observe that language model has no effect for the Nouns task. On the contrary, on the verbs task filtering already significally improves performance, while combining language model scores with initial paradigm probabilities increases prediction quality by 3 percents more. It is easy to explain since the main source of errors for nouns was the confusion between animate/inanimate nouns where both the predictions are phonologically plausible. Conversely, in the Verbs task the mispredicted forms in imperative like **осуществься* has low probability according to character ngram models which allows the system to exclude them.

The main contribution of our paper is corpora-based algorithm for paradigm prediction. Again, we accomodate corpora counts together with the logarithmic probabilities predicted by the basic classifier on the second stage of our algorithm. More precisely, after generating the word forms $w_1, \ldots, w_m$

of the lexeme L according to hypothetic paradigm c_j, we calculate the corpus score by the formula $C = \sum_{j=1}^{m} -\log C(w_j)$, where $C(w_j)$ is the number of times w_j occurs in the corpora. All counts are incremented by 1 to avoid zero probabilities. This method resembles the method of [Ahlberg et al., 2014,], however, we make one modification to deal with homonymy: if a word form occurs two times in the paradigm (for example, in nominative and genitive), then we divide all the corpora counts of it by 2. Without this modification, this algorithm favours invariable nouns.

However, we are still unable to discriminate between unanimate and animate nouns by our algorithm since the set of word forms is the same in both cases. The only difference is that genitive forms of animate nouns would be more frequent than the ones of unanimate since they appear in accusative also. To capture this difference we should measure the similarity between the expected distribution of case forms and the observed proportion of their counts. Let $\mathcal{P} = [p_1, \ldots, p_m]$ be the expected probabilies of different word forms according to their grammemes and $\mathcal{N} = [N_1, \ldots, N_m]$ be their observed counts. We normalize the empirical distribution by its sum $N = \sum_j N_j$, obtaining the empirical probability distribution $\mathcal{Q} = [q_1, \ldots, q_m]$ where $q_j = \frac{N_j}{N}$. Then the difference score equals

$$D(\mathcal{N}, \mathcal{P}) = \sum_j q_j \log \frac{q_j}{p_j} \cdot \log N$$

Note that this measure is simply Kullback-Leibler divergence between $\mathcal{Q}$ and $\mathcal{P}$ multiplied by the log count of the given lexeme. The expected form counts were collected in the training phase separately for each paradigm. The results for corpora-based paradigm prediction are shown in Table 10. We used the counts from Russian National Corpora available on `ruscorpora.ru/corpora-freq.html`.

Task	No corpora	Corpora counts as features	Counts and divergences as features
Nouns	77.38 93.50	80.21 95.34	82.73 95.67
Verbs	76.30 88.83	84.30 93.81	83.66 93.73

Table 10: Using character model for paradigm prediction

We observe that using corpora counts indeed leads to a substantial gain in performance in both tasks. However, in the case of verbs most of the

advantage is obtained from corpora counts themselves, using similarity scores slightly worsens performance. On the Nouns task similarity scores, on the contrary, leads to a further improvement in per-table accuracy. Indeed, the most difficult problem for nouns is animacy/unanimacy differentiation where absolute counts are useless. In the verb tasks, conversely, homonimy plays no role, therefore, similarity scores are redundant and make the data more noisy.

Inspecting remaining incorrect predictions, we found that in the Verbs task they are mainly caused by wrong imperative form generation. Often corpus counts cannot resolve this problem because imperative forms are not very frequent for many verbs: both *кровоточи* and **кровоточь* do not appear in the RNC counts. Often corpora features are not powerful enough to overcome the gap caused by first level classifier. For example, for the verb *лгать* the correct paradigm has probability 0.01 after the first stage. Joint classifier raises it up to 0.3, however, it is too low to rank this hypothesis on the top. The same problem arises in the task of noun paradigm prediction: for most of the erroneous predictions the correct paradigm was excluded already by the basic classifier or obtained an extremely low probability.

We also combined character ngram scores with the corpora-based classifier, which improved the performance further. For the Nouns task the gain was marginal (82.80% instead of 82.73% for per-table accuracy), however, the accuracy of paradigm prediction for verbs achieved 86.51% instead of 84.30%. The per-form accuracy also increased significantly, reaching 95.66% in comparison with 93.81%.

6 Conclusion

We have developed a system for automatic paradigm induction and prediction. Our algorithm of paradigm induction is based on the method of longest common subsequence. To predict paradigms automatically we apply a logistic regression classifier using suffix and prefix features. This classifier achieves accuracy of 77% on Russian nouns and 76% on Russian verbs in paradigm prediction task, the percentage of correctly predicted forms is 93% and 88% respectively. We have also designed a corpora-based algorithm of paradigm prediction using the basic classifier on its first stage. This improves the accuracy of paradigm prediction to 81% on nouns and 84% on verbs, per-form accuracy reaches 95 and 93%. These results are substantially better then previously achieved for Russian in [Ahlberg et al., 2015] (the authors of that work used another dataset and experiment setting).

We plan to improve our results further by using corpora information more extensively. Our results show that taking into account relative frequencies of grammemes enhances the quality of corpora-based methods. Therefore modelling the distribution of grammemes more accurately should leave to further improvement. For this goal we plan to use morphologically disambiguated corpora. Another improvement could be achieved by grouping together the corpus statistics for the words of presumably the same paradigm.
Our results could be used for automatic morphological analysis and synthesis in such tasks as POS-tagging or lemmatization. Modern techniques of lemmatization such as used in [Jonjejan, Dalianis, 2009] also use the LCS approach but apply it to each word form separately without using full inflectional table. Our method incorporates information from the whole paradigm, therefore it could potentially improve state-of-the-art algorithms of morphological analysis for Russian. Since our system does not predict the best inflection table only, but returns the probabilities of possible paradigms, it can be used as a component of a joint classifier, taking into account context model probabilities as well as single word scores. Using context information together with suffix/prefix features could also help to determine word part-of-speech, which is a preliminary step for our algorithm.
This task is especially important for Web texts, which contain numerous out-of-vocabulary words whose inflection cannot be determined by dictionary-based methods. We plan to test our approach for morphological processing of social media texts in future studies.

References

[Ahlberg et al., 2014] Ahlberg M., Forsberg M., Hulden M. (2014) Semi-supervised learning of morphological paradigms and lexicons // EACL 2014, p. 569.

[Ahlberg et al., 2015] Ahlberg M., Forsberg M., Hulden M. (2015) Paradigm classification in supervised learning of morphology // Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics Human Language Technologies (NAACL-HLT 2015), Denver, CO, pp. 1024–1029.

[Chen,Goodman, 1996] Chen S. F., Goodman J. (1996) An empirical study of smoothing techniques for language modeling // Proceedings of the 34th annual meeting on Association for Computational Linguistics, pp. 310–318.

[Durrett, DeNero, 2013] Durrett G. and DeNero J. (2013) Supervised Learning of Complete Morphological Paradigms. // HLT-NAACL, pp. 1185–1195.

[Fan et al., 2008] Rong-En Fan R.-E., Chang K.-W., Hsieh C.-J., Wang X.-R., Lin C.-J. (2008) LIBLINEAR: A Library for Large Linear Classification // Journal of Machine Learning Research, Vol. 9, pp. 1871–1874.

[Jonjejan, Dalianis, 2009] Jongejan B., Dalianis H. (2009) Automatic training of lemmatization rules that handle morphological changes in pre-, in-and suffixes alike. // Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP, Vol. 1, pp. 145–153.

[Lyashevskaya,Sharoff, 2009] Lyashevskaya O. and Sharoff S. (2009) Frequency dictionary of modern Russian language [Chastotnyj slovar' sovremennogo russkogo yazyka], Azbukovnik, Moscow.

[Nicolai et al., 2015] Nicolai G., Cherry C., Kondrak G. (2015) Inflection Generation as Discriminative String Transduction // Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics Human Language Technologies (NAACL-HLT 2015), Denver, CO, pp. 923–931.

[Padro, Stanilovsky, 2012] Padro L., Stanilovsky E. (2012) FreeLing 3.0: Towards Wider Multilinguality // Proceedings of the Language Resources and Evaluation Conference (LREC 2012), Istanbul, Turkey, pp. 2473–2480.

[Pedregosa et al., 2011] Pedregosa F., Varoquaux G., Gramfort A., Michel V., Thirion B., Grisel O., Blondel M., Prettenhofer P., Weiss R., Dubourg V., Vanderplas J., Passos A., Cournapeau D., Brucher M., Perrot M., Duchesnay E. Scikit-learn: Machine Learning in Python (2011). // Journal of Machine Learning Research, Vol. 12., pp. 2825-2830.

[Zaliznyak, 2002] Zaliznyak A. A. (2002) Russian nominal inflection with a supplement of selected works on modern Russian and general linguistics. [Russkoe imennoe slovoizmenenie s prilozheniem izbrannyh rabot po sovremennomu russkomu yazyku] Yazyki slavianskoj kul'tury, 2002.

7 Appendix

No	Abstract paradigm	Count	Example
1	1#1+ы#1+а#1+ов#1+у#1+ам #1#1+ы#1+ом#1+ами#1+е#1+ах	959	0=аборт, 1=аборт
2	1+е#1+я#1+я#1+й#1+ю#1+ям #1+е#1+я#1+ем#1+ями#1+и#1+ях	622	0=Евангелие, 1=Евангели
3	1+а#1+ы#1+ы#1#1+е#1+ам #1+у#1+ы#1+ой#1+ами#1+е#1+ах	444	0=автомашина, 1=автомашин
4	1+ь#1+и#1+и#1+ей#1+и#1+ям #1+ь#1+и#1+ью#1+ями#1+и#1+ях	330	0=активность, 1=активност
5	1+я#1+и#1+и#1+й#1+и#1+ям #1+ю#1+и#1+ей#1+ями#1+и#1+ях	270	0=авария, 1=авари
6	1#1+ы#1+а#1+ов#1+у#1+ам #1+а#1+ов#1+ом#1+ами#1+е#1+ах	249	0=абонент, 1=абонент
7	1+2+а#1+2+и#1+2+и#1+о+2 #1+2+е#1+2+ам#1+2+у#1+2+и #1+2+ой#1+2+ами#1+2+е#1+2+ах	239	0=арка, 1=ар,2=к
8	1#1+и#1+а#1+ов#1+у#1+ам #1#1+и#1+ом#1+ами#1+е#1+ах	222	0=аналог, 1=аналог
9	1#1+и#1+а#1+ов#1+у#1+ам #1+а#1+ов#1+ом#1+ами#1+е#1+ах	174	0=академик, 1=академик
10	1+о#1+а#1+а#1#1+у#1+ам #1+о#1+а#1+ом#1+ами#1+е#1+ах	143	0=агентство, 1=агентств

Table 11: Most frequent abstract paradigms for Russian nouns

No	Abstract paradigm	Count	Example
1	1+ть#1+ю#1+ешь#1+ет#1+ем#1+ете#1+ют #1+л#1+ла#1+ло#1+ли#1+й#1+йте	1316	0=арестовывать, 1=арестовыва
2	1+ться#1+юсь#1+ешься#1+ется, #1+емся#1+етесь#1+ются#1+лся, #1+лась#1+лось#1+лись#1+йся#1+йтесь	568	0=барахтаться, 1=барахта
3	1+овать#1+ую#1+уешь#1+ует #1+уем#1+уете#1+уют#1+овал #1+овала#1+овало#1+овали#1+уй#1+уйте	302	0=агитировать, 1=агитир
4	1+ить#1+ю#1+ишь#1+ит#1+им#1+ите #1+ят#1+ил#1+ила#1+ило#1+или#1+и#1+ите	192	0=благодарить, 1=благодар
5	1+ить#1+у#1+ишь#1+ит#1+им#1+ите #1+ат#1+ил#1+ила#1+ило#1+или#1+и#1+ите	117	0=вершить, 1=верш
6	1+ить#1+лю#1+ишь#1+ит#1+им#1+ите #1+ят#1+ил#1+ила#1+ило#1+или#1+и#1+ите	116	0=благословить, 1=благослов
7	1+иться#1+юсь#1+ишься#1+ится #1+имся#1+итесь#1+ятся#1+ился #1+илась#1+илось#1+ились#1+ись#1+итесь	104	0=валиться, 1=вал
8	1+дить#1+жу#1+дишь#1+дит #1+дим#1+дите#1+дят#1+дил #1+дила#1+дило#1+дили#1+ди#1+дите	89	0=бродить, 1=бро
9	1+оваться#1+уюсь#1+уешься#1+уется#1+уемся #1+уетесь#1+уются#1+овался#1+овалась #1+овалось#1+овались#1+уйся#1+уйтесь	71	0=адаптироваться, 1=адаптир
10	1+уть#1+у#1+ёшь#1+ёт#1+ём#1+ёте#1+ут #1+ул#1+ула#1+уло#1+ули#1+и#1+ите	66	0=блеснуть, 1=блесн

Table 12: Most frequent abstract paradigms for Russian verbs