Extending phpMorphy with dialect words

Maxim L Gromov, Svetlana S Zemicheva
Tomsk State University, Lenin av. 36, 634050 Tomsk, Russia

Abstract. This paper describes the work done to create a tool for adding new (dialect) words into the dictionary of phpMorphy. PhpMorphy is a tool written in PHP which helps to annotate a word with grammatical information. It was chosen because it needs not to be trained and it can be integrated into PHP based text corpora search system relatively simply. The search is planned to be done over inverse index of words. The search requests are supposed to contain desired grammatical description of words (eg, “find all feminine adjectives”, etc.). For that reason inverse index should contain grammatical information. The process of adding new words into phpMorphy dictionary technically is not hard however it is time consuming and tedious. To facilitate this work the mentioned tool was developed. Also paper describes the process how the words to add into the phpMorphy dictionary were selected. The words from the given dialect dictionary were run through phpMorphy. If phpMorphy correctly recognizes a word, then it should not be added into the phpMorphy dictionary. Otherwise it should be.

1. Introduction

The creation of dialectal corpora is one of the important tasks of modern linguistics, because it meets the need to reflect the synchronous variability of the language, including its oral variety. However, the development of this kind of resources requires significant efforts at the initial stage (field work, decoding of oral speech). The most famous is the dialect subcorpus of the National Corpus of the Russian Language [1], which, despite the presence of different dialect types (North Russian, Central Russian, South Russian) cannot be considered representative for the whole country (it includes 285281 tokens from 22 regions only). An alternative to the consolidated corpus of Russian dialects is the creation of corpora of separate territories. The concept of such resource was proposed by Saratov linguists [2], however, it was not fully implemented. As of this moment, there are dialectal corpora of several Russian regions: Ustja river basin corpus (Arkhangelsk region) which includes 1195150 tokens [3, 4], corpus of Rogovatka dialect (Belgorod region, 114600 tokens) [5], corpus of the Russian dialect spoken in the village Malinino (Lipetsk region, 114600 tokens) [6], electronic corpus of the dialect culture of Kuban [7], the electronic textual corpus of the linguistic culture of the Northern Angara region (170813 tokens) [8], Tomsk dialect corpus (1560199 tokens) [9–11], dialect corpus of the Middle Irtysh Region [12] and others. In some of these projects only a thematic markup is presented; morphological one is not even planned.

The number of morphologically annotated dialect corpora made on the material of the Russian language is extremely small. These include the aforementioned dialect subcorpus as part of the National Corpus of the Russian Language and Ustja river basin corpus. Moreover, in all cases, fully automatic marking of the morphology with the help of standard tools is used (the developers of the Ustja river basin corpus use TreeTagger [4], the developers of the dialect subcorpus of the National...
Corpus of the Russian Language use MyStem), which generates a significant number of errors. In the dialect subcorpus as part of the National Corpus, all materials after automatic processing undergo manual disambiguation. Thus, their standard of morphological markup includes special categories ("dialect contraction form", "dialect suffix", "dialect stem"), as well as the filter "dialect features" in each grammatical category, which allows the user to find examples for studying a particular phenomenon of dialect morphology quickly. [13].

Among the tools that work with grammatical information, there is a freely distributed tool phpMorphy [14]. PhpMorphy is chosen because of two reasons. Firstly, phpMorphy is written in PHP and therefore its integration into our online search system (also written in PHP) over dialect corpora is native and relatively simple. Secondly, other systems (eg., TreeTagger and myStem) need to be retrained for our purposes, and this requires a corpus, annotated with grammatical information, which we do not have. PhpMorphy, in other hand, does not need training; it basically relies on its dictionary. However, there are no dialect words in the internal dictionary of phpMorphy. But there are a number of published dialect dictionaries and it’s possible to use them for extraction base forms of dialect words as well as some grammatical information. Our goal is to create a system of morphological markup for the Tomsk dialect corpus, and the dialect dictionary of the Middle Ob region was used.

The dictionary was published as a three-volume edition in the 60s of the twentieth century [15], later two additional volumes were published [16, 17]. At the first stage of the project, the dictionary was scanned, dictionary entries from different volumes were automatically combined into a common alphabetical list, and proofreading was carried out: errors in automatic text recognition were fixed. The dictionary contains more than 20000 lexemes. Among them there are dialecticisms of different types: phonetic (анбăр – амбаř), morphological (калош – masculine, singular, nominative, “overshoe”), semantic (картошка – standard meaning “potato”, in the dialect denotes георгин “dahlia”), derivational (летова'ть – проводить лето “spend summer”) as well as words with dialect roots (алюсничать – сплетничать “to gossip”) and affixes (деревнюшка – diminutive from “village”). Obviously, semantic dialecticisms will be mostly processed without mistakes and other types can cause troubles.

This article focuses on the development of a tool for adding new words into the phpMorphy dictionary. The development was divided into two stages. The first stage (Section II) is to find out which dialect words are already in the phpMorphy dictionary, which are correctly guessed by phpMorphy (that is, phpMorphy finds the correct basic form of the word), and which are not in the phpMorphy dictionary (and are not guessed right). Next (Section III) the tool itself was developed.

2. Terms filtering
PhpMorphy allows restoring main and all other forms of a given word. If the word (or rather its inflexible part) is present in the internal dictionary of phpMorphy, the word is called found. If a word is not found in the dictionary, phpMorphy tries to guess what word it is (what part of speech) and in what form it is. If the latter succeeds, the word is called guessed. PhpMorphy does not try to guess the found words. The word may be not found, and not guessed.

Any form of a guessed or found word is accompanied by grammatical information describing that form (for example, a masculine noun in the genitive case, animate).

Before adding new words to the phpMorphy dictionary, it was necessary to find out which words are already present in the dictionary or correctly guessed, and to concentrate efforts on adding to the dictionary only those words that are absent from the dictionary and are not guessed (or guessed, but incorrectly).

2.1. Filtering Before PhpMorphy
The presence of the dialect dictionary and its structure make it possible to extract certain grammatical information from it. Firstly, each dictionary entry includes word base form (nouns are in the nominative case, singular; verbs are in the infinitive, etc.).
Secondly, the word definition and the grammatical description of the word are stored in the same string. In most cases, the grammatical information is separated from the definition of the word by a dot. Grammatical information is given at the beginning of the string; its various items (flections in the genitive case, gender for nouns, flections of the feminine and neuter genders for adjectives, etc.) are listed separated by commas. A dot is placed only before the beginning of the definition. Splitting the string by the first dot, one can get two pieces of the original string: the first part – the grammatical description, the second – the definition of the word. But sometimes the grammatical description itself contains a dot. It happened because the dictionary was compiled manually by different people at different times and the form of the grammatical information was not unified. In this case splitting the string by the first dot gives a “truncated” grammatical description.

The descriptions themselves are in most cases standard and can be formalized (for example, “а, cp” – the flection of the word in the genitive case is “а”, the word itself is a neuter noun), but sometimes there are deviations. It is clear during manual processing, a person could accurately determine what is meant. But processing should be done automatically. Therefore the words with unclear or incomplete grammatical information were discarded. Dictionary entries such as collocations, two-letter words, and those including a reference to other entries (with the foot-note “see there”) were also discarded. From 24,792 words 4,620 words were discarded, i.e. for further consideration remained 20,172 words.

2.2. Filtering Using phpMorphy

The words selected from the dialect dictionary were run through phpMorphy one by one. At the same time we used the fact that the words in the dictionary are in their basic forms.

There were 4 possible outcomes when the word ran through phpMorphy. The word was not found and was not guessed (as the result the word was rejected). The word was found or guessed, but among the proposed forms of the word there was no one corresponding to the grammatical information specified in the dialect dictionary (again, the word was rejected). The word was found in the dictionary and among the forms of the word there was a form with the necessary grammatical information (the word was marked as found). And the word was guessed and among the forms of the word there was a form with the necessary grammatical information (the word was marked as guessed).

It should be noted, that found and guessed words were rejected not only when the grammatical information did not coincide with the one from the dictionary, but also when the base form of the word provided by phpMorphy did not coincide with the base form of the word from the dictionary (in fact, with the word itself). For example, the dictionary contains a word “ВЫШЕНЬ noun feminine nominative case” (with the meaning “top, peak”). If we run it through phpMorphy, phpMorphy finds it and grammatical information is as should be (“noun feminine nominative case”). But the phpMorphy version of spelling of the base form (“ВЫШНЯ”) does not coincide with the dictionary version (“ВЫШЕНЬ”). For that reason the word should be rejected.

2.3. Result of filtering

Among the words selected for phpMorphy 5249 were found in the phpMorphy internal dictionary, 8620 words were guessed right by phpMorphy and 6303 words were rejected.

Thus, in total there are 5249 + 8620 = 13869 words guessed or found, and 4620 + 6759 = 10923 words (discarded in the first stage or rejected in the second) unknown for phpMorphy.

3. An instrument for adding new items into phpMorphy dictionary

3.1. Structure of the phpMorphy Dictionary Source

The internal phpMorphy dictionary is stored in several binary files. However, phpMorphy includes a system for building binary dictionaries from the dictionary source code. The source code of the dictionary is a fairly simple xml file. There are five main branches in this xml file:

- parts of speech – poses;
- items of grammatical information –(grammems);
ancodes, each ancode is a pair of the form (part of speech; list of grammems);
models of word construction (flexias), each model is a triple of the form (a prefix, which should be added at the beginning of a stem to get a word; a flexia, which should be added at the ending of a stem to get a word; ancode that describes the resulting word (part of speech and a list of grammems));
lemmas, each lemma is a pair of the form (stem; a model of word construction).

Naturally, when an object (for example, a lemma) uses another object (for example, a model of word construction), it is not the object used but its identifier.

Obviously, new words (lemmas) should be added to the dictionary source code to the lemmas branch. This can be done manually, one just should specify a stem and choose the right model, but it is very time-consuming. The number of different flexias in the current state of the phpMorphy dictionary for the Russian language is 2765. Therefore, it was decided to create a tool that would facilitate this work.

3.2. Selecting flexia model

When designing the tool, the following use case scenario was kept in mind. A user types a stem and selects a model of word construction from the list of models. The list of words obtained as a result of application of the selected model to the stem is automatically displayed. Each word is accompanied by grammatical information that is embedded in the model. If the user is satisfied with the list, he or she presses the button to add the resulting lemma into the dictionary.

However, this scenario does not free the user from viewing all 2765 models of word construction. The list should be reduced somehow. The following method was chosen. For each part of speech from all possible forms the most characteristic forms were selected. The user specifies the suffixes (prefixes are not used in this version of the tool) that should be added to the stem to get a particular form. Among all models, only those are searched, that correspond to the specified forms (contain grammatical information accompanied with the suffixes), and only these models are displayed for the user to select. After looking through the found models and selecting the most suitable one, the user clicks the button to add the lemma to the dictionary source code.

The forms chosen for nouns are: nominative singular, genitive singular, accusative singular and nominative plural.

The forms chosen for adjectives are: nominative masculine singular, nominative feminine singular, nominative neuter singular and nominative plural.

The forms chosen for verbs are: active voice present 1st person singular, active voice past tense masculine singular, active voice imperative 2nd person plural.

The forms chosen for participles are: active voice present singular masculine nominative and the passive voice present singular masculine nominative.

The rest parts of speech are not considered so far.

3.3. Process of adding new lemmas

Lemmas are added to the dictionary locally. That is, each user of the tool has a local copy of the xml file. This approach was chosen because organizing a centralized repository of the dictionary would require extra effort. There would be a need to allocate additional network resources, create an access control system, a system for synchronization issues and arbitration of simultaneous access to the dictionary, etc. These tasks would require additional resources to create a tool and most likely would increase a production time, which was unacceptable. We believe that the chosen “local approach” is justified, because the number of users of the tool (at least for our projects) is planned to be no more than 10 people, and such a small team can always agree on who processes what part of the dictionary, and how to combine xml-files in the future.

The process of adding new lemmas is divided into two stages in the tool. In the first stage, the user adds lemmas to the intermediate file. In the second stage, the system transfers lemmas from the intermediate file to the xml file. This is done because the process of modifying the xml file is a long operation due to the large size of the xml file.
3.4. Implementation
The tool is implemented in Python. The graphical user interface was built using Qt. The appearance of the tool window is shown in Figure 1. The tool is designed as a single thread application, which is not the best solution because of the long operation that “hangs up” the interface for a while. However since this operation is supposed to be done at the end of a work session, the simplicity of a single threaded design is a good compromise. The tool is free. Its source code can be downloaded from GitHub [18].

![Figure 1. The main window of the tool](image)

4. Conclusion and future works
In this paper we have described a part of works of the large project to development of electronic resources for the study of the folk speech culture of the Middle Ob region. The content of these works was to check which words of the dictionary of dialect words of the Middle Ob region are known (guessed or found) to phpMorphy, and which are not. The unknown words should be added to the internal dictionary of phpMorphy. To add new words (lemmas) to phpMorphy dictionary a tool was developed.

As a future work on the tool, we plan to implement extra functionality for adding new models of a word construction (flexias). There is a need for this, because it may happen that none of the 2765 models is suitable for some very specific dialect word.

When all unknown words are added to the phpMorphy dictionary, phpMorphy will be used to create an inverse index of the text corpus. This index is planned to contain words together with their grammatical description. This will make possible to implement search through the text corpus for words, words with grammatical information specified or just grammatical information.

In addition, information about the lexical and grammatical classes of words conveying various connotations (diminutive, augmentative, etc.) can be extracted from the dialect dictionary and then used as an element of the semantic markup of the dialect corpus.

Acknowledgments
The work presented in this paper was supported by the Russian Science Foundation, Project No. 19-78-10015 “Development of electronic resources for the study of folk speech culture of the Middle Ob” (2019–2022).

References
[1] National corpus of the Russian language URL http://www.ruscorpora.ru/search-dialect.html
[2] Kriuchkova O Yu and Goldin V E 2011 A corpus of Russian dialectal speech: the concept and parameters of evaluation [Korpus russkoy dialektnoy rechi: kontseptsiya i parametry otsenki] Computational Linguistics and Intellectual Technologies: papers from the Annual International Conference on “Dialogue” (2011) 10 pp 359–67 (in Russian)
[3] Basic stats URL http://parasolcorpus.org/Pushkino/stats.php
[4] Waldenfels R, Daniel M and Dobrushina N 2014 Building the Ustya River Basin Corpus, an online corpus of a Russian dialect Computational Linguistics and Intellectual Technologies: papers from the Annual International Conference “Dialogue” (2014) 13 pp 270–8
[5] Corpus of Rogovatka dialect URL http://www.parasolcorpus.org/Rogovatka/
[6] Corpus of the Russian dialect spoken in the village Malinino URL https://linghub.ru/malinino/#!/
[7] Tregubova E N 2015 Multilevel thematic marking as an ethnolinguistic tool of dialectal discourse representation in digital text corpora [Mnogourovnevaya tematicheskaya razmetka kak instrument etnolingvisticheskoy reprezentatsii dialektnogo diskursa v elektronnom tekstovom korpusе] Tomsk State University J. of Philology 1 (33) pp 66–77 (in Russian)
[8] Electronic text corpus of linguoculture of the Northern Angara region [Elektronnii tekstovyi korpus Severnogo Priangarya] URL http://angara.sfu-kras.ru/?page=dialect# (in Russian)
[9] Tomsk dialect corpus [Tomskii dialektnyi korpus] URL http://losl.tsu.ru/corpus (in Russian)
[10] Yurina E A 2011 Tomsk dialect corpus: the starting point [Tomskij dialektnyj korpus: v nachale puti] Tomsk State University J. of Philology 2 (14) pp 58–63 (in Russian)
[11] Zemcheva S S, and Ivantsova E V 2018 The project of Tomsk dialect corpus in keeping with trends of corpus linguistics development [Proyekt sozdaniya Tomskogo dialektnogo korpusa v svete tendentsiy razvitiya korpusnoy lingvistiki] Siberian J. of Philology 3 pp 192–205 (in Russian)
[12] Lavrov D N, Lapin A P, Kharlamova M A and Cherkashchenko I A 2019 Text Processing with Phonetic and Thematic Markup for Displaying Texts of Dialect Corpus [Obrabotka teksta s foneticheskoy i tematicheskoy razmetkami dlya otobrazheniya tekstov dialektnogo korpusa] Mathematical structures and modeling 3 (51) pp 135–144 (in Russian)
[13] Kachinskaya I B and Sichinava D V 2017 On the corpus of dialectal texts in the Russian national corpus [O korpusе dialektnykh tekstov v nacionalnom korpusе Russkogo yazyka] Russian J. of Lexicography 11 pp 71–85 (in Russian)
[14] phpMorphy URL http://phpmorphy.sourceforge.net
[15] Palagina V V, Blinova O I, Yantsenetskaya M N, Sokolov O M and Ivanova F P 1964–67 The Dictionary of the Russian Old-Timer Dialects of the Central Basin of the Ob River [Slovar’ russkih starozhil’cheskikh govorov Sredney chastii basseyna r. Obi] vol 1–3 (Tomsk, Russia Tomsk State University Publ., Tomsk) (in Russian)
[16] Zakharova L A, Sypchenko S V et al. 1975 The Dictionary of the Russian Old-Timer Dialects of the Central Basin of the Ob River (supplement) [Slovar’ russkihstarozhil’cheskikh govorov Sredney chastii basseyna r. Obi (dopolnenie)] Pt 1, 2 (Tomsk: Tomsk State University Publ.) (in Russian)
[17] 1983–1986 The Middle Ob Dictionary (supplement) [Sredneobskiy slovar’ (dopolnenie)] vol 1, 2 (Tomsk: Tomsk State University Publ.) ed. Palagina V V (in Russian)
[18] Gromov M 2019 Add lemma to phpMorphy URL https://github.com/maxim-leogromov/add_lemma_phpmorphy