Bi-LSTM Model for Morpheme Segmentation of Russian Words

Elena Bolshakova[1] , Alexander Sapin[2]

[1] Lomonosov Moscow State University,
National Research University Higher School of Economics, Moscow, Russia
eibolshakova@gmail.com,
[2] Lomonosov Moscow State University, Moscow, Russia
alesapin@gmail.com

Abstract. The paper addresses the task of automatic morpheme segmentation involving both splitting words into morphs and classification of resulted morphs. For segmentation of Russian words, a new model based on Bi-LSTM neural network is proposed and experimentally evaluated on several training data sets differing in labeling. The proposed model has comparable quality with the best supervised machine learning models for morpheme segmentation with classification, slightly outperforming them in word-level classification accuracy with score 89% .

Keywords: morphological segmentation, morpheme analysis of Russian words, neural network models for morphology, morpheme segmentation with classification

1 Introduction

Morpheme segmentation is a kind of morphological analysis, which is less studied than traditional morphological tagging. The task of automatic morpheme segmentation involves splitting words into constituent morphs, the surface forms of morphemes (roots and affixes), for example: *soul-ful*, *Rus. за-душ-евн-ый*. Morphemes are the smallest meaningful units of texts, so information about morphemic structure of words is useful for various NLP problems, in particular, in lexical semantics to overcome data sparseness inherent to natural languages. The work [3] shows that even simple subword information can improve distributional word vectors representations, therefore more accurate linguistic information about word structures is helpful for deriving meaning of rare and out-of-vocabulary words. Morphemic structure of words is also exploited in such tasks as recognition of semantically related words: cognates, paronyms and so on.

The problem of data sparseness is more significant for languages with rich morphologies (e.g., Russian or Finnish) with many affixes (prefixes, suffixes, postfixes) of various types and meanings, and at the same time, for such languages morpheme

parsing of words is especially difficult. For morphology rich languages, a more complicated task is topical, which requires not only morpheme segmentation but also classification of segmented morphs by labeling their types (the main types are Prefix, Root, Suffix, Ending), for example: *за*:PREF/*душ*:ROOT/*евн*:SUFF/*ый*:END, *soul*:ROOT/*ful*:SUFF.

The first works on morpheme segmentation were pure statistical [2, 8], but significant progress was achieved by machine learning techniques [1, 7, 11-13]. Initially, only small amount of annotated words was available for training, so unsupervised and semi-supervised methods were firstly developed for morpheme segmentation, without classification of resulted morphs. The most known solutions of the task are implemented in Morfessor system [7].

The work [11] presents a semi-supervised method for morpheme segmentation based on conditional random fields (CRF) and sequential labeling of letters. However, the task of morpheme segmentation with classification remained unexplored until recent works [5, 13] undertaken for Russian, due to appearance of relevant labeled data. These works have applied powerful machine learning techniques to the task of segmentation with classification of segmented morphs, resulting in the following models with open-source code:

- Convolutional neural network model[1] (CNN) [13], which was trained on the data obtained from an electronic version of Tikhonov's dictionary [15];
- Gradient boosted decision trees model[2] (GBDT) [5] trained and evaluated both on data from CrossLexica system [4] and the data of Tikhonov's dictionary.

These supervised models classify letters of words according to the main types of morphs and achieve F-measure about 98% for morpheme boundaries (outperforming unsupervised Morfessor's methods) and also they show morph classification accuracy for whole words up to 88% , presenting state-of-the-art for the considered task.

Since various neural network models demonstrate an evident progress for most NLP tasks, we aimed to further research machine learning methods for morpheme segmentation with classification and to apply one more neural model suitable to tasks of sequence labeling [14], namely bidirectional long short-term memory neural network (Bi-LSTM) [9]. A similar recurrent neural network (in combination with CRF method) was already exploited in [12] for word segmentation in several languages but not Russian.

In the paper we describe the created Bi-LSTM model[3] for Russian and report results of its evaluation and comparison with two above-mention models of morpheme segmentation with classification. For training and evaluating our model, we exploited two available data sets having different labeling of morphemes. The first dataset contains about 90,000 segmented words of Tikhonov's dictionary [15], and the second, about 23,000 words taken from CrossLexica [4]. We have also used extended data obtained by converting and merging these two data sets.

[1] https://github.com/AlexeySorokin/NeuralMorphemeSegmentation

[2] https://github.com/alesapin/GBDTMorphParsing

[3] https://github.com/alesapin/RussianMorphParsing

Experimental evaluation has shown that for morpheme segmentation our Bi-LSTM model is comparable with the above-mentioned state-of-the-art machine learning models, and it slightly outperforms them in word-level classification accuracy, giving score 89%.

The paper starts with a brief overview of the main works on morpheme segmentation, followed by explaining two available data sets for training, as well as our approach to merge them by machine learning. Then key features of our Bi-LSTM model are described, and results of experiments with it are reported and discussed. Finally, we present some conclusion.

2 Related Work

The first known method of morpheme segmentation was proposed by Z. Harris [8], it detects morph boundaries at locations with local maximums of letter frequencies counted for various positions in the words. For English, the method showed only 61% of precision, however Harris' letter variety statistics was then used in many subsequent works.

A more precise method for morpheme segmentation was developed in Morfessor [7] and based on unsupervised machine learning with a large unlabelled text. A semi-supervised improvement of the method (by exploiting some segmented data) was implemented in Morfessor 2.0, showing F-measure on morph boundaries about 77-80% for English, Finnish, and Turkish.

Supervised machine learning for morpheme segmentation was proposed in the work [11] and based on conditional random fields (CRF). The task was considered as sequential labeling for letters of a given word, by assigning each letter to one of three classes (beginning or middle of a multi-character morph or a single character morph). The developed CRF classifier makes use of Harris' values as features of a letter being classified. To overcome low-resource learning (only 1,000 segmented words) it also exploits features obtained by Morfessor (through unsupervised learning), thus implementing semi-supervised learning and increasing F-measure on morph boundaries to 84-91% for English, Finnish, and Turkish.

A pure supervised method for morpheme segmentation task (but without classification of segmented morphs) was developed in [1] with sequence-to-sequence neural network (Seq2seq) and trained for Russian on the data of Tikhonov's dictionary [15], demonstrating 93.95% of F-measure for detecting morpheme boundaries.

The next supervised model [13] also exploits Tikhonov's dictionary and applies convolutional neural networks (CNN) for the twofold task of morpheme segmentation with classification, for Russian. The task is considered as sequence labeling by classifying letters with 22 classes based on BMES labeling scheme. In order to recognize successive suffixes, prefixes, roots in words with complex structure, the classes account for beginning (B), middle (M), and ending (E) positions of a letter in the corresponding root, affix or word ending, as well as single (S) letter variants, linking letter in multi-root words (*вод-**о**-лаз*), and also hyphen in hyphenated words (*по-*

хорошему). Here is an example of labeling taken from [13], for the word *учитель* (*teacher*) segmented as *уч*:ROOT/*и*:SUFF/*тель*:SUFF :

у	*ч*	*и*	*т*	*е*	*л*	*ь*
B-ROOT	E-ROOT	S-SUFF	B-SUFF	M-SUFF	M-SUFF	E-SUFF

The CNN model is quite complicated and involves post editing of predicted classes by an auxiliary correcting procedure, which fixes some wrong sequences of classes (in particular, if a postfix is encountered within a root). The model outperforms the previously developed models for morpheme segmentation with F-measure up to 98% and also achieves classification accuracy about 96% for letters and 88% for whole words.

Besides the described supervised models of morpheme segmentation for Russian, a rule-based approach was proposed in [10]. However, even in combination with statistical information, the approach significantly loses to supervised ones (its best result is about 85% of F-measure for detecting morpheme boundaries in Russian words).

The recent paper [5] presents one more supervised model of morpheme segmentation with classification for Russian, which is based on decision trees with gradient boosting (GBDT). Unlike the CNN model, the number of classes was reduced to 10 (since the set of BMES labels is redundant for the considered task). The GBDT classifier takes into account features of a letter being classified (in particular, its position in the word and Harris' values), features of its word (some morphological tags: part of speech, case, etc.), and also 5 previous and 5 subsequent letters. The model was separately trained both on the data of Tikhonov's dictionary and the data of CrossLexica system and then compared with the other known models trained on the same data. Overall, the quality of GBDT model turned out to be comparable with CNN model.

Therefore, it seems reasonable to attempt to improve their results with a neural network of another type, as well as to study the impact of data sets merging.

3 Data Sets for Training

For now, there are at least two data sets with Russian words spitted into morphs: the data of Tikhonov's dictionary [15] and a subset of CrossLexica's dictionary [4]. In both data sets segmented morphs are classified according main types of Russian morphemes (prefix, root, suffix, ending, postfix), and successive prefixes and suffixes are separated. The sets are suitable for training, but they have the following significant differences, some of them are explained by origin and purposes of the dictionaries, as well by the fact that there is no full agreement between linguists about correct splitting Russian words into morphs (and also about classification of some morphemes).

- Size of the data sets: 96,046 words in Tikhonov's dictionary *Vs.* 23,426 words of CrossLexica.
- Differences in lexicon: Tikhonov’s dictionary contains many obsolete words (e.g., *рапуха*, *отъять*), while CrossLexica does not include such words, but contains some modern words (e.g., *эксклюзивный – exclusive*).
- Complexity of word structure: unlike Tikhonov’s dictionary, CrossLexica's data does not encompass multi-root words and words with hyphen.

- Different classification of some morphems: for example, in Tikhonov's data set word *успеть* (*to be in time*) is segmented as *успе*:ROOT/*ть*:SUFF while CrossLexica's variant is *у*:PREF/*спе*:ROOT/*ть*:END (affix *ть* of verb infinitive may be interpreted either as suffix or ending).
- Differences in splitting words into morphs: unlike Tikhonov's dictionary, where many prefixes and suffixes are not separated from roots (because of their semantic cohesion), in CrossLexica all affixes are usually segmented, in particular, the separated prefix *у* in the previous example of the word *успеть*; here is one more example: *пространство* (*space*) is segmented as *пространств*:ROOT/*о*:END in Tikhonov's data, while in CrossLexica's data the word has another segmentation: *про*:PREF/*стран*:ROOT/*ств*:SUFF/*о*:END.

The last discrepancy is evidently a consequence of different dictionary destinations: Tikhonov's dictionary is a derivational dictionary, while the words of CrossLexica were segmented and labeled for purposes of constructing morpheme paronyms (words with the same root but different affixes and so having close meanings, such as *массивный* and *массовый – massive* and *mass*).

Similar to work [5], we have trained and evaluated our Bi-LSTM morheme segmentation model separately on both these data sets, but additionally we have merged the datasets, in order to study data increment effect on model quality.

The merging was performed in two following ways. First, CrossLexica's data set was converted into Tikhonov's format of segmenting and labeling and then added to Tikhonov's 's data (we denote the obtained dataset as TN+CL'). The second way is vise versa: Tikhonov's data were converted into CrossLexica's format of labelling and added to CrossLexica's data (the resulted data set is denoted as CL+TN'). The second way seemed a more perspective because of more consistent labeling presented in CrossLexica.

To implement converting, we applied machine learning, specifically, a relatively simple neural network model with two Bi-LSTM [9] layers and one fully connected softmax layer, taking for training segmented and labeled words common for the original data sets (i.e. words present in both datasets, their number is 15,639). Since CrossLexica's data has no hyphenated and multi-root words, for relevancy of the conversion Tikhonov → CrossLexica we have added to the training set 1,768 multi-root words and 989 words with hyphen, they were manually annotated. The obtained sets were randomly divided in proportion 80:20 for training and testing of the corresponding conversion models.

Our choice of Bi-LSTM model for data conversion relies on the same reasons, as the choice of Bi-LSTM for morpheme segmentation, since converting may be considered as a letter sequence labeling task. The quality of the trained models for conversion (and so quality of converting) is presented in Table 1, segmentation is evaluated as precision, recall and F1-meagure on morph boundaries, while classification, as accuracy both for letters and for whole words. One can see that in both cases the results are quite precise, and conversion Tikhonov → CrossLexica is turned more reliable.

Thus, we had four data sets for training our Bi-LSTM developed for morpheme segmentation with classification: two original datasets and two merged ones.

Table 1. Evaluation of data sets conversion

Conversion of labeling: Source set → Goal set	Segmentation			Classification	
	Precision	Recall	F-measure	Letter accuracy	Word accuracy
CrossLexica →Tikhonov	97.05	97.60	97.32	97.03	85.29
Tikhonov → CrossLexica	97.99	98.26	**98.25**	**98.25**	**90.51**

4 Model Architecture

Recurrent neural networks are applied in various NLP tasks, and long short-term memory (LSTM) neural network [9] as their kind, is useful for sequences labeling tasks [14]. Since in morpheme segmentation with classification class of a letter may depend on some subsequent and previous letters, the core of our model is bidirectional LSTM (Bi-LSTM). Our Bi-LSTM model was implemented with Keras library[4] [6] (based on Tensorflow).

As model input we use letters represented in one-hot encoding format, complementing them with morphological tags of the word being segmented (POS, case, gender, number, tense), which are taken from morphological parser CrossMorphy[5], one-hot encoded and concatenated with letter vectors.

To align all words to the same length, we evidently exploit padding, but with masking residual letters <PAD> (by excluding them while calculating errors), in order to avoid their influence on gradient descent.

Preliminary experiments with Bi-LSTM model showed that the best results are obtained with multiple layers, so we stack layers with dropout between them. Similar to [14] we use skip connections between layers, which transfer data from two previous layers to the next one with concatenation, and this accelerates learning and improves model quality. The last layer of the model is fully connected and completed with softmax activation function which outputs probability distribution over all possible letter classes.

Various parameters of the Bi-LSTM model were experimentally tested, for all experiments the data sets were randomly divided in proportion 80:20 for training and testing, respectively. We did not use validation data subsets, relying on the following considerations. Our data sets are big enough, with a wide variety of morphemic word structures, and cross-validation takes a lot of time (training our Bi-LSTM network on GPU is about six times slower than for CNN model). But which is more important, we were aimed to reveal only such hyper parameters of our Bi-LSTM model, as the number of layers, the number of units in each layer, and dropout, so the chance of overfitting for randomly formed training and testing data subsets is negligible. Additionally, we tested our model on several random splits and obtained approximately the same results.

[4] https://github. com/fchollet/keras

[5] http:/gihub.com/alesapin/XMorphy

The main optimal hyper parameters turned out to be the following. The model has three LSTM layers and dropout of 12%. The first layer includes 768 units and others, 512 units. Among the gradient descent algorithms (adam, RMSprop, SGD), better results were shown by RMSprop. Due to skip connections, the number of training epochs is guaranteed no more than 55, and also the number of epochs is limited by the early-stop algorithm, which stops training after 10 epochs if the quality is not improving. The LSTM layers were implemented with CuDNNLSTM layer, which is trained faster on GPU, but has fixed activation function (tahn) and cannot perform recurrent dropout.

Similar to CNN and GBDT models for morpheme segmentation, the output of our Bi-LSTM model is corrected by an auxiliary procedure that relies on simple rules of morphotactic for Russian words: any word should begin with a prefix or root, a root can go after prefix, a suffix can go after root or another suffix, and so on. The resulted classes of letters are obtained from probability distribution with argmax function.

5 Results of Experiments and Comparison

We have separately trained our Bi-LSTM model on four above-described datasets (two original and two merged ones), thus obtaining four trained models. For training all models, instead of full BMES labeling scheme with 22 classes of letters, which was exploited in [13] for CNN model, we apply the limited set of 10 classes proposed in [5] for GBDT model, since this is sufficient for the task. We should note, that for training and evaluating the models on merged datasets (TN+CL' and CL+'TN) we have exploited random splittings with testing subsets containing only examples (4683 segmented words) from original sets (TN and CL, respectively).

All trained Bi-LSTM models were evaluated by precision, recall and F1-meagure on morph boundaries (for segmentation), and also accuracy both for letters and whole words (for classification). The results of evaluation are shown in Table 2. For a word to be accurately classified, all the morph boundaries in it should be correct and classes of all its letters must be true.

Table 2. Evaluation of Bi-LSTM models for segmentation with classification

Data Sets	Segmentation			Classification		Corrected words (%)
	P	R	F	Letter accuracy	Word accuracy	
TN (Tikhonov)	97.66	97.88	97.77	96.11	86.41	2.91
CL(CrossLexica)	98.68	98.83	98.76	98.00	93.31	**0.25**
TN+CL'	98.29	97.97	98.13	96.77	88.38	2.17
CL+TN'	**98.80**	**99.00**	**98.90**	**98.18**	**93.68**	3.09

The best results for all applied measures belong to the model trained on CrossLexica's dataset extended by converted Tikhonov's dictionary (CL+'TN), this can be easily explained by the absence of multi-root words and words with hyphen in the training data set. For the same reason the model on purely CrossLexica's data has the second

results. Both models trained on extended data slightly outperforms the models based on original datasets, however, overall, scores of all the models are highly close. Thus, data merging does not really influence the resulting model quality, despite not quite accurate data conversion. This means that our approach to data converting is applicable for extending the model without losing precision.

The last column of Table 2 presents percentage of errors corrected by the auxiliary procedure based on rules of morphotactic and intended to fix wrong morpheme types in the outputs of the machine learning models. One can see that the ratio of fixed errors is quite low for all the models. It should be noted that it is less than analogous ratio of fixed errors showed by the state-of-the-art models of morpheme segmentation with classification: 4-10% for convolutional neural network (CNN) and 0.2-4% (depending on training datasets) for gradient boosted decision trees (GBDT) model (cf. [5]). Therefore, purely for recognizing morphs in words (without post correction), Bi-LSTM model seems more relevant than GBDT and CNN models.

Table 3. Comparison of classification accuracy for the best segmentation models

Model	CrossLexica's Data Set		Tikhonov's Data Set	
	Letters	Words	Letters	Words
CNN	97.88	93.23	96.64	88.71
GBDT	**98.39**	94.20	96.40	86.54
LSTM	98.33	**94.49**	**96.89**	**89.03**

We also compared the results of our Bi-LSTM model with the analogous results of CNN and GBDT models trained on Tikhonov's and CrossLexica's data set, their scores were taken from [5, 13]. For correct comparison with CNN model, we have trained ensemble of three Bi-LSTM models, since the best score for CNN model was obtained for such an ensemble with averaging the results. For all the compared models, we evidently have used the same splitting of the data sets for training and evaluating.

Table 3 shows comparison of three models by their classification accuracy: the scores are close, but our model demonstrates slightly better word-level accuracy (the scores for segmentation are omitted because they are even closer, about 98% of F1-measure for all models).

Finally, we have estimated ratio of various errors, depending on wrong boundaries between morphemes of various types:

- Successive prefixes (PREF_PREF), such as wrong segmentation for word *осложнение:* *ос*:PREF/*ложж*:ROOT/*н*:SUFF/*ени*:SUFF/*е*:END (in CrossLexica's model) instead of correct variant *о*:PREF/*с*:PREF/*ложж*:ROOT/*н*:SUFF/*ени*:SUFF/*е*:END;
- Prefix and root (PREF_ROOT), e.g., erreneous *озим*:ROOT/*ый*:END and correct о:PREF/зим:ROOT/ый:END for *озимый* (CrossLexica's model);
- Successive roots (ROOT_ ROOT): wrong *пере*:ROOT/*прав*:ROOT/*а*:END instead of *переправ*:ROOT/*а*:END for *переправа* (Tikhonov's model);

- Root and suffix(ROOT_SUFF), such as *вареж*:ROOT/*к*:SUFF/*и*:END instead of correct *варежк*:ROOT/*и*:END for *варежка* (Tikhonov's model);
- Successive suffixes (SUFF_SUFF): *мешк*:ROOT/*ов*:SUFF/*ин*:SUFF/*а*:END instead of *мешк*:ROOT/*овин*:SUFF/*а*:END for *мешковина* (CrossLexica);
- Suffix and ending (SUFF_END): wrong *неш*:ROOT/*и*:SUFF/*е*:END instead of *неш*:ROOT/*ие*:END for word *нешие* (Tikhonov's model);
- the other errors, such as wrong recognition of postfix.

Table 4. Types of errors in morpheme segmentation (%)

Data Sets	PREF_ PREF	PREF_ ROOT	ROOT_ ROOT	ROOT_ SUFF	SUFF_ SUFF	SUFF_ END	Other
Tikhonov	2.62	22.18	11.60	**48.32**	11.43	2.05	1.8
CrossLexica	7.36	24.81	3.88	**36.43**	25.97	1.55	0

The error scores for the Bi-LSTM models trained on the original datasets are presented in Table 4. In both models, the most frequent errors are related with wrong boundaries between roots and suffixes (almost half of the errors for Tikhonov's data). Another frequent error is wrong recognition of roots (for Tikhonov's data) or erroneous segmentation of suffixes (for CrossLexica's data). It seems that correct recognition of roots is more crucial for NLP applications than identification of boundaries between suffixes, so the model trained on CrossLexica's data is more preferred though does not account for multi-root words (but they have relatively low frequency).

6 Conclusion and Future Work

We have developed and evaluated a Bi-LSTM model for twofold task: morpheme segmentation of Russian words and classification of segmented morphs, exploiting for training the model data sets with different labeling of morphs. Four trained Bi-LSTM models have shown comparable results with the other best machine learning models, namely GBDT (the gradient boosted decision trees) and CNN (convolutional neural network), and thus each of them can be used in various NLP experiments with Russian text, the choice of the model depends on particular application.

Since such different machine learning models have comparable quality, we relate further progress in precision of morpheme segmentation and classification of segmented morphs with improvement of data sets exploited for training. For now, the available data sets suffer from some inconsistencies and errors. So we plan to work out rules to improve data consistency and to study other ways to merge labeled data, with subsequent validation of all segmentation models.

At the same time, our Bi-LSTM model slightly outperforms the others in word-level accuracy and needs to correct less errors. It is also interesting to try the model for another morphologically rich language.

Acknowledgements. We would like to thank the anonymous reviewers of our paper for their helpful and constructive comments.

References

1. Arefyev, N. V., Gratsianova, T. Y., Popov, K. P.: Morphological Segmentation with Sequence to Sequence neural network. In: Computational Linguistics and Intellectual Technologies: Proceedings of the Int. Conference "Dialogue 2018", Moscow, pp.82–91 (2018)
2. Bernhard, D.: Simple morpheme labelling in unsupervised morpheme analysis. In: Workshop of the Cross-Language Evaluation Forum for European Languages. Springer Berlin Heidelberg, pp. 873-880 (2007)
3. Bojanowski, P., Grave, E., Joulin, A., Mikolov, T. : Enriching word vectors with subword information. Transactions of the Association for Computational Linguistics, 5, pp. 135-146 (2017)
4. Bolshakov, I.A.,.: CrossLexica – Universum of links between Russian words. Business Informatics, No 3 (25), pp.12–19 (2013) (in Russian)
5. Bolshakova, E. I., Sapin, A.S.: Comparing models of morpheme analysis for Russian words based on machine learning: In: Comput. Linguistics and Intellectual Technologies: Papers from the Annual Int. Conference "Dialogue 2019", Moscow, pp. 104-113 (2019)
6. Chollet, F.: Keras: Deep learning library for theano and tensorflow (2015), https://keras.io/
7. Creutz, M., Lagus, K.: Unsupervised models for morpheme segmentation and morphology learning. ACM Transactions on Speech and Language Processing, 4(1), Article 3 (2007)
8. Harris S. Zellig: Morpheme boundaries within words: Report on a computer test. Transformations and Discourse Analysis Papers 73, pp. 68–77 (1967)
9. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation, 9(8), pp. 1735-1780 (1997)
10. Maltina, L., Malafeev, A.: Automatic Morphemic Analysis of Russian Words. In: CEUR Workshop, Vol. 2268, pp. 85-94. (2018)
11. Ruokolainen, T., et al. : Painless semi-supervised morphological segmentation using conditional random fields. In: Proceedings of the 14th Conference of the European Chapter of the ACL, Vol. 2: Short Papers, pp. 84-89 (2014)
12. Shao, Yan: Cross-lingual Word Segmentation and Morpheme Segmentation as Sequence Labelling. In: First Workshop on Multi-Language Processing in a Globalizing World (MLP 2017), Dublin, Ireland. arXiv preprint arXiv:1709.03759 (2017)
13. Sorokin, A., Kravtsova, A. : Deep Convolution Networks for Supervised Morpheme Segmentation of Russian Language. In: Proceedings of the Conference on Artificial Intelligence and Natural Language, St-Petersburg, Springer, Cham, pp. 3–10 (2018)
14. Sutskever, I., Vinyals, O., Le, Q. V.: Sequence to sequence learning with neural networks. Advances in neural information processing systems, pp. 3104-3112 (2014)
15. Tikhonov, A.N.: Word Formation Dictionary of Russian language, Moscow, Russkiy yazyk (1990)