Using longest common subsequence and character models to predict word forms

Alexey Sorokin
Moscow State University / GSP-1, Leninskie Gory, 1
Faculty of Mathematics and Mechanics, 119991, Moscow, Russia
Moscow Institute of Physics and Technology / Institutskij per., 9,
Faculty of Innovations and High Technologies, 141701, Dolgoprudny, Russia
alexey.sorokin@list.ru

Abstract

This paper presents an algorithm for automatic word forms inflection. We use the method of longest common subsequence to extract abstract paradigms from given pairs of basic and inflected word forms, as well as suffix and prefix features to predict this paradigm automatically. We elaborate this algorithm using combination of affix feature-based and character ngram models, which substantially enhances performance especially for the languages possessing nonlocal phenomena such as vowel harmony. Our system took part in SIGMORPHON 2016 Shared Task and took 3rd place in 17 of 30 subtasks and 4th place in 7 substasks among 7 participants.

1 Introduction

Automatic paradigm detection has attracted extensive attention in recent years. The most valuable works include (Ahlberg et al., 2014; Ahlberg et al., 2015), which pursue a classification-based approach, encoding every inflection paradigm by a single label, and (Dreyer and Eisner, 2011; Nicolai et al., 2015) applyng transduction-base techniques. The representation of morphological paradigms in (Ahlberg et al., 2014) is based on the *longest common subsequence* (LCS) method, suggesting that the common material in the LCS is the "stem" and the symbols not in the LCS are inflection markers. This approach actually goes back to the 60s and was first applied in the seminal work of Andrey A. Zaliznyak "Russkoe imennoe slovoizmenenie" ("Russian nominal inflection", Zaliznyak (2002)). From the lingustic point of view, transduction-based techniques are more relevant, since when inflection is realised by several simultaneous modifications (say, prefix and suffix changes and vowel alterations in the stem), it is more natural to represent it by multiple operations, not by a single label. Nevertheless we decide to use the first approach: firstly, machine learning algorithms used for classification are more simple computationally than the ones for string transformations and require less parameter tuning. Secondly, our work was initially conducted for Russian, where most morphological alterations occur in affixes with the exception of several patterns. Our method is based on the one of Ahlberg et al. (2015) with several technical modifications.

Our system participated in SIGMORPHON-2016 Shared Task on morphological inflection (Cotterell et al., 2016). This task required one to guess a single word form of a given category, not the whole paradigm, which is a reverse for one of the standard NLP tasks: lemmatization, where the goal is to predict a basic form of the word (the lemma) given its inflected form. Lemmatization is rarely considered as labeling task, the only exception being Gesmundo and Samardžić (2012). However, the latter work considers only alterations at the edges of the word, while the LCS method allows one to address internal changes as well. A more general method of edit trees used in Chrupala et al. (2008) and Müller et al. (2015) is another realization of the LCS approach. However, lemmatization algorithms extensively exploit context to discriminate between candidate forms which is irrelevant for SIGMORPHON challenge.

Our paper consists of 5 parts, including the introduction. The second part briefly explains how abstract paradigms are induced from data. It also introduces the techniques used for their automatic detection. The third part outlines SIGMORPHON SHARED TASK and the methods we proposed to solve it. The fourth part describes different vari-

Proceedings of the 14th Annual SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology, pages 54–61,
Berlin, Germany, August 11, 2016.

ants of our system and compares their relative performance. We conclude with the error analysis and future work directions.

2 Abstract paradigms

A standard task of paradigm learning is to guess an inflection table given a source form. In the classification approach, this task is solved by encoding the complete table with a label and then predicting this label using standard machine learning technique. The predicted values in our method are *abstract paradigms* introduced in Ahlberg et al. (2014). Informally speaking, an abstract paradigm is a list of patterns, where common parts of word forms are replaced by variables. These variables correspond to maximal contiguous segments of the longest common subsequence of these forms. For example, paradigms *sing-sang* and *drink-drank* both share the same abstract paradigm **1+i+2#1+a+2**. The same approach can be applied not only to complete paradigms, but also to the pairs (source form, inflected form).

Following Ahlberg et al. (2014), we use finite automata to extract LCS. However, our implementation has several technical differences, mentioned below. When several variants of the same length exist, we extract the one with the least total number of gaps and the least total length of these gaps. These two quantities being equal, the number of zero-length gaps is minimized. For example, given Arabic verb *imtāza* "to be distinguished" and its inflected form *tamtaz*, the best LCS is *mt-z* , not *mt-a* since it has two gaps of length 1 instead of one gap of length 2 and one of length 0. We also restrict the maximal length of gaps in the middle and in the beginning of the word to prevent spurious matches. When the language lacks prefix inflection, we simply set the maximal initial gap to 0.

After enumerating all possible abstract paradigms, the task of automatic inflection becomes a standard classification problem. Our basic system uses lemma suffixes and prefixes as features, following the scheme of Ahlberg et al. (2014). We bound their length by some number (usually 5 for suffixes and 3 for prefixes) and encode them as binary features. We also remove frequent affixes before extracting the features, attaching them afterwards. Since the number of possible features can reach several thousands, we perform extensive feature selection. For the competition we enriched the model by character ngram scores which is our main contribution and significantly boosted performance. Details are given in Section 4.

3 Sigmorphon Shared Task

First SIGMORPHON SHARED TASK (Cotterell et al., 2016) was held for 10 languages of different genealogy and morphological structure. There were several agglutinative languages (Finnish and Turkish), two Semitic languages (Arabic and Maltese) with "root-and-pattern" morphology, some languages are more difficult to characterize (Navajo). We refer the reader to the organizers' article for a complete description of the task. The competition consisted of 3 parts. In the first participants were asked to construct a word form given its lemma and morphological tag. In the second task the source form was not required to be the lemma, but its tag was also given. In the third task the tag of the source form was unknown.

In the first task our system was trained for each morphological tag separately. Every lemma-form pair was treated as an abstract paradigm containing two cells. Given a lemma at test time, the classifier uses its suffixes and prefixes to detect the most probable abstract paradigm. A paradigm being known, we determine the variable values. Substituting this values for the variables in the second cell of the paradigm, we calculate the target form. The scheme of the process described is presented in Figure 1.

In the second task we first map the form of the source category to its lemma, training on the reverse pairs from the first task, then apply a direct classifier from Task 1. Since our system assigns probabilities, for every source form w_1 of the category c_1 we calculate a set of possible lemmas $l_1, \dots, l_m$ with their probabilities; afterwards for every lemma we construct its possible forms of target category c_2 and select the forms with the highest total probability $P(w_2|w_1, c_1, c(w_2) = c_2) = \sum_i p(w_2|lem(w_2) = l_i, c(w_2) = c_2)p(lem(w_1) = l_i)$. The algorithm is depicted on Figure 2.

For the third task we learn the transformation from the word form to its lemma, using the reversed training data for Task 1 with target category omitted (only the POS label is known). Then the obtained lemmas were transform to word forms of target category just as in Task 2 using the same

Source form (pos=V,lemma)	features	paradigms (with probs)		variables	target form (pos=V,mood=SBJV, per=3,tense=PRS, num=SG, ...)
detentar	ˆd, ˆde	1+ar#1+e	0.82	1=detent	detente
	r$, ar$, tar$	1+e+2+ar#1+ie+2+e	0.13	1=det, 2=nt	detiente
	ntar$, entar$	1+ar#1+ue	0.05	1=detent	detentue

Figure 1: How we model inflection (Task 1). For each lemma in the test set we predict an abstract paradigm, which consists of lemma and target form patterns. Fitting the lemma to its guessed pattern, we find the variables in the paradigm. Using these variables, the target word form is constructed.

Source form pos=V,polar=POS, mood=IMP,per=1,num=PL	Lemmas (with probs)		Target forms (for each lemma)		Target forms (with probs) pos=V,mood=IND,tense=PST, per=1,num=SG,aspect=PFV	
vaguemos	vagar	0.83	vagué	0.90	vagué	0.92
			vagé	0.10	vagé	0.08
	vaguar	0.17	vagué	0.97		
			vaguué	0.03		

Figure 2: How we model reinflection (Task 2). First, for the source form a list of possible lemmas is constructed. Second, for each potential lemma all possible target forms are predicted. The probabilities of these forms are calculated using chain rule.

probability formula. Our system essentially relies on 2 basic components: forward transformation from lemma to its inflected form and the corresponding backward transformation. Therefore in what follows we focus on implementation issues and evaluate performance mainly for these two problems.

We deliberately refrain from using any external sources including linguistic information and corpus statistics. Nevertheless, our method could be easily extended with corpora features, see Ahlberg et al. (2014) and Sorokin and Khomchenkova (2016) for possible strategies.

4 Performing classification

In this section we present some details of our system used to solve Task 1 and its reverse, for the general description see Section 2. Each inflection/lemmatization task was solved without any information about other forms of the given lemma. We applied a logistic regression classifier using suffixes and prefixes as features. Usually the length of suffixes was 5 for inflection and 6 for lemmatization, the maximal length of prefix features was 3. However, prefix features were not used for Turkish and Finnish, while for Navajo we used prefixes of length up to 5 and the length of suffixes was bounded by 3. In all languages words were classified by their last (first for Navajo) letter and a separate classifier was trained for each letter. To avoid data sparsity we performed extensive feature selection, keeping only best 10% of features according to an ambiguity measure (**?**) and disregarding features observed less than 3 times in the training set. If an affix of length up to 3 unambigiously predicts the paradigm label π, we assign π for any lemma ending by (beginning with) this affix. We also experimented with removing frequent suffixes before extracting features and attaching them afterwards, which slightly improved the results for most of the languages. Language-dependent parameters are given in the Appendix.

We report the results of our system evaluation in Table 1, column SIMPLE. We used the train-dev split provided by the workshop organizers after removing all the duplicates from the development set.[1] For most of the tasks it outperforms the baseline which uses the transductive approach with an averaged perceptron as the classifier. However, for

[1]Actually, it was not done in the submission version, which caused slight overfitting for several tasks.

Language	Verbs			Adjectives			Nouns		
	JOINT	SIMPLE	BASE	JOINT	SIMPLE	BASE	JOINT	SIMPLE	BASE
Arabic	80.9	66.0	65.5	94.4	87.2	63.2	76.2	73.4	76.3
Finnish	94.0	93.4	58.4	62.9	62.9	14.3	87.8	87.2	77.9
Georgian	42.3	32.1	48.7	100.0	100.0	100.0	97.6	96.6	95.3
German	90.0	89.3	85.5	97.2	96.7	91.0	91.2	91.2	89.2
Hungarian	92.5	89.8	78.7				75.9	71.4	58.0
Navajo	94.5	93.4	84.6				56.4	47.9	53.9
Russian	83.2	82.8	81.2	95.8	95.8	95.8	91.9	91.5	91.9
Spanish	98.6	98.6	96.2	100.0	100.0	99.1	100.0	100.0	99.5
Turkish	83.5	74.4	61.2				87.3	78.4	56.3

Table 1: Performance quality for inflection (Task 1).

Russian and Arabic there is a marginal gap, while for Georgian and Navajo verbs our results are not satisfying enough.

We try to close this gap using character models. We learn an ngram model on the set of word forms in the train data. Model counts are smoothed using Witten-Bell smoothing. We integrate ngram probabilities in our model via standard reranking scheme: the SIMPLE model is used to generate the n-best list. Then for every form in this list we calculate its log-probability according to SIMPLE model as well as language model logarithmic score normalized by word length. These two features are passed to a logistic regression classifier and the hypothesis with highest decision score is returned. To learn the weights of the scores we apply the following procedure from Joachims (2002): suppose three hypotheses w_1, w_2 and w_3 were generated for the lemma l in the training set , w_2 being the correct word form. If their log probabilities are p_1, p_2, p_3 and n-gram log scores are s_1, s_2, s_3 respectively, we include in the training set vectors $[p_2 - p_1, s_2 - s_1]$ and $[p_2 - p_3, s_2 - s_3]$ as positive instances and $[p_1 - p_2, s_1 - s_2]$ and $[p_3 - p_2, s_3 - s_2]$ as negative. Note that this ranking scheme allows for integration of other task-dependent features as well which might be helpful in lemmatization or POS-tagging.

The results of the improved system are also given in Table 1, column JOINT. We observe that for most of the tasks the combined system confidently beats the baseline, except Georgian verbs. Arabic nouns and Russian nouns and adjectives are on par with the baseline system. Character ngram models are the most helpful for the languages with developed vowel harmony, such as Turkish (the impact for Finnish is more modest since the performance quality of the SIMPLE system was already high). In case of Arabic SIMPLE model often generates about ten forms of approximately the same probability; the character model helps to select the best one and rule out the words which do not satisfy local phonological requirements.

Table 2 contains results for the reverse task of lemmatization used as the first stage in Tasks 2 and 3. There was no baseline available, therefore we compare only performance of the SIMPLE system and the JOINT system using character ngram models. We observe that ngram statistics produce more substantial gain in this task than for inflection. We also provide evaluation results on Tasks 2 and 3 of the Shared Task (Table 3, first line for each language stands for Task 2 and second for Task 3). Since accuracy for these tasks is determined by lemmatization and inflection quality, we will not discuss them further. For the results of the competition itself we refer the reader to organizers' article (Cotterell et al., 2016), we just mention that our system was ranked the 3rd 17 times of 30, 5 times for Task 1 and 6 times for Tasks 2 and 3.

5 Error analysis and discussion

A morphological inflection system may fail to recover a correct form for two reasons: first, it is too restricted to generate a correct paradigm, second, its features are not strong enough to discriminate between correct and incorrect labels. To distinguish these two cases we calculated the recall of our system both for the inflection and lemmatization subtasks, measuring whether a correct word receives probability larger than 1%. Results are collected in Table 4. Interestingly, the ranking precision (which is the percentage of cases when the

Language	Verbs		Adjectives		Nouns	
	JOINT	SIMPLE	JOINT	SIMPLE	JOINT	SIMPLE
Arabic	76.1	56.7	94.5	93.6	83.1	63.6
Finnish	92.7	85.0	62.9	48.7	90.7	84.7
Georgian	51.3	34.6	99.2	98.4	99.4	96.6
German	94.3	92.1	98.1	96.8	94.1	90.7
Hungarian	98.7	96.1			99.1	98.2
Navajo	65.6	44.2			64.2	52.2
Russian	87.5	82.8	97.1	96.8	93.6	89.7
Spanish	98.5	96.5	100.0	99.1	97.2	98.2
Turkish	93.8	91.3			97.0	96.4

Table 2: Performance quality for lemmatization.

Language	Verbs		Adjectives		Nouns	
	JOINT	BASE	JOINT	BASE	JOINT	BASE
Arabic	81.9	55.0	86.9	61.3	71.0	62.8
	84.3	50.9	86.9	49.0	69.2	46.5
Finnish	88.0	74.3	55.6	5.6	89.7	52.5
	88.0	51.1	52.8	5.6	87.5	71.0
Georgian	46.9	37.0	96.0	97.0	97.5	94.2
	44.4	40.6	94.9	92.6	98.1	94.2
German	86.2	81.0	97.8	91.4	91.0	87.2
	85.7	74.4	97.6	87.6	88.8	70.6
Hungarian	95.0	79.9			77.8	63.0
	96.3	78.8			75.0	63.0
Navajo	54.5	47.6			100.0	89.0
	56.6	56.8			100.0	81.1
Russian	78.9	71.7	95.9	96.2	89.5	89.0
	77.1	70.1	96.9	91.6	89.1	83.9
Spanish	98.0	95.5	100.0	96.6	97.2	95.7
	97.9	91.5	100.0	70.3	97.6	84.3
Turkish	85.2	56.0			88.8	54.0
	86.4	55.2			87.7	51.6

Table 3: Performance quality on Tasks 2 and 3.

correct word form was ranked the first provided it was in the candidate list) for the overwhelming majority of tasks is over 90% which shows that affix features and character models are indeed effective in discriminating between candidate paradigms. Omitting Georgian verbs and Finnish adjectives, where the classifier suffers from the lack of training data, we observe two problematic languages: Arabic demonstrates decent recall and moderate precision in both tasks, while results on Navajo degrade mostly due to extremely low recall except the lemmatization of nouns, where precision drops to 66%.

As a key example, consider the *Pres+2+Sg+Masc* form of Arabic verbs, for which the percentage of correctly predicted forms was only 62% (8 of 13). In 4 of 5 erroneous cases the algorithm fails to even suggest the correct transformation (**1+ā+2+a#ta+1+ū+2+u**, e. g. *dāma* “to last” – *tadūmu* “you (Masc) last”) because it was never observed in the training set for this particular task and was observed only once at all. The fact that in other forms *ā* was often replaced by *ū* also does not help since transformations are considered “as a whole”, not as a sequence of local edits. The remaining mistake (*taḍraḡiṭṭu* instead of *taḍriḡaṭṭu* for *iḍraḡaṭṭa* “to leave”) is also frequent: the algo-

Language	Verbs			Adjectives			Nouns		
	Recall	Accur.	Prec.	Recall	Accur.	Prec.	Recall	Accur.	Prec.
Arabic	88.2	80.4	91.16%	95.9	93.6	97.60%	83.9	76.2	90.82%
	85.5	76.1	89.01%	96.1	94.5	98.34%	84.4	83.1	98.46%
Finnish	95.1	94.1	98.95%	63.9	62.9	98.44%	96.0	87.8	91.46%
	96.1	90.6	94.28%	65.7	62.9	95.74%	98.0	92.7	94.59%
Georgian	59.8	42.3	70.74%	100.0	99.2	99.2%	98.5	97.6	98.09%
	62.9	51.3	81.56%	100.0	100.0	100.0%	99.8	99.4	99.60%
German	93.3	90.0	96.46%	98.1	97.2	99.08%	94.8	91.2	96.20%
	94.3	94.3	100.0%	98.4	98.1	99.70%	96.6	94.1	97.41%
Hungarian	97.5	92.5	94.87%				82.1	75.9	92.45%
	99.1	98.7	99.60%				99.1	99.1	100.0%
Navajo	61.8	56.4	91.26%				97.8	94.5	96.63%
	67.9	64.8	95.43%				95.6	63.3	66.21%
Russian	91.4	83.2	91.03%	98.5	95.8	97.26%	98.0	91.9	93.78%
	88.2	86.6	98.19%	97.7	95.2	97.44%	96.8	94.1	97.21%
Spanish	98.8	98.6	99.80%	100.0	100.0	100%	100.0	100.0	100%
	98.6	98.5	99.90%	100.0	100.0	100%	99.5	97.2	97.69%
Turkish	87.7	83.5	95.21%				89.5	87.3	97.54%
	93.8	93.8	100.0%				96.8	96.4	99.59%

Table 4: Recall and precision on inflection (upper) and lemmatization (lower) tasks.

rithm correctly predicts the paradigm description **i+1+a+2+a#ta+1+i+2+u** but erroneously replaces the first root syllable instead of the second. In the case of ambiguity the position of alteration is determined by calculating the probability of "local" transformation **a** $\rightarrow$ **i** in both locations and choosing the most probable according to its immediate context. Since local features cannot distinguish first and second syllables of the root and paradigm description contains no information on the number of vowels in variables, our system cannot avoid such mistakes. However, their percentage can be lowered by allowing the system to predict several surface variants for one abstract paradigm (in current architecture this decision is made before calculating ngram scores).

Another difficult problem for our approach is fusion. Consider the *REAL+Pl+1* form of the Navajo verbs, where only 2 predictions of 7 are correct and in 5 remaining cases the correct paradigm label was not even in the list of possible paradigms. For example, the plural form *deiijįį'* of the verb *yiijįįh* "to stand up" is produced using the pattern **y+1+h#de+1+'** while the training set contained only the transformation **y+1+h#dei+1+'** for *yik'ęęh* → *deiik'ęę'*, where the prefix *de-* was fused with initial *y*. That explains the low performance on Navajo inflection in terms of recall. The opposite problem holds for lemmatizing Navajo nouns. For example, all 4-th person forms start with *ha-*, however, it can be obtained from the lemma form either by adding initial *h* (*ataa'*-*hataa'*) or by substitutiing for initial *bi-* (*bijaa'*-*hajaa'*). This ambiguity cannot be resolved with dictionary or corpus; fortunately, when reinflecting other forms all the potential lemmas generate the same form so the performance on Tasks 2 and 3 is unaffected.

Summarizing, our classification-based approach meets difficulties when faced with 'too local' phenomena like fusion of 'too global' like vowel harmony. Indeed, there is no way to predict vowels in the middle of the word observing only its edges. This difficulty is resolved using character ngrams, which can capture the interdependency between nearby vowels in the word stem. Using models of order up to 6 significantly improves performance on Arabic, Turkish and Finnish. When applying ngram models in the lemmatization stage we observe consistent improvement practically for all languages. Character models cannot help when a correct paradigm was not generated as a candidate

[1] Generally, our system was ranked higher on Tasks 2 and 3 which means that lemmatization part works better than inflection partially due to character ngram usage

(recall the discussion on Arabic verbs above). There are two possible strategies in this case: first, a deterministic postprocessing step could be applied to "harmonize" vowels or make other phonetic transformations. Another variant is to create with every abstract paradigm its "phonetic variants" and perform the classification step on the extended set of paradigms. We plan to address this question in future research.

The last thing to say, one of the important merits of our system is its simplicity. It does not require complex techniques for parameter tuning; training the model also relies on well-established algorithms. The features we use are also very simple and they could be easily extended, for example, to capture the context information. Taking into account the solid performance of our system in SIGMORPHON SHARED TASK, we think that classification-based approaches could be useful in a couple of tasks including paradigm induction, morphological parsing and lemmatization for languages of arbitrarily complex morphological structure.

References

Markus Dreyer and Jason Eisner. 2011. Discovering morphological paradigms from plain text using a dirichlet process mixture model. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 616–627. Association for Computational Linguistics.

Thomas Müller, Ryan Cotterell, Alexander Fraser, and Hinrich Schütze. 2015. Joint lemmatization and morphological tagging with Lemming. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, Lisbon, Portugal*, pages 2268–2274. Association for Computational Linguistics.

Alexey Sorokin and Irina Khomchenkova. 2016. Automatic detection of morphological paradigms using corpora information. In *Dialogue. 22nd International Conference on Computational Linguistics and Intellectual Technologies*, pages 604–616, Moscow, Russia, June. RSUH.

Malin Ahlberg, Markus Forsberg, and Mans Hulden. 2014. Semi-supervised learning of morphological paradigms and lexicons. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 569–578, April.

Malin Ahlberg, Markus Forsberg, and Mans Hulden. 2015. Paradigm classification in supervised learning of morphology. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics Human Language Technologies (NAACL-HLT 2015), Denver, CO*, pages 1024–1029, June.

Garrett Nicolai, Colin Cherry, and Grzegorz Kondrak. 2015. Inflection generation as discriminative string transduction. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics Human Language Technologies (NAACL-HLT 2015), Denver, CO*, pages 923–931, June.

Andrea Gesmundo and Tanja Samardžić. 2012. Lemmatisation as a tagging task. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers-Volume 2*, pages 368–372. Association for Computational Linguistics.

Grzegorz Chrupala, Georgiana Dinu, and Josef van Genabith. 2008. Learning morphology with Morfette. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, pages 2362–2367, Marrakech, Morocco, May. European Language Resources Association (ELRA).

Ryan Cotterell, Christo Kirov, John Sylak-Glassman, David Yarowsky, Jason Eisner, and Mans Hulden. 2016. The SIGMORPHON 2016 shared task—morphological reinflection. In *Proceedings of the 2016 Meeting of SIGMORPHON*, Berlin, Germany, August. Association for Computational Linguistics.

Thorsten Joachims. 2002. Optimizing search engines using clickthrough data. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 133–142. ACM.

Andrey A. Zaliznyak. 2002. *Russian nominal inflection*. Yazyki slavyanskoj kultury, Moscow, Russia.

Acknowledgements

The author thanks Irina Khomchenkova for careful assistance during the research. I am also very grateful to anonymous ACL SIGMORPHON reviewers whose detailed comments were very helpful in improving this paper. This work is a part of General Internet Corpus of Russian (`http://www.webcorpora.ru/`) project.

Appendix

Language	Max. gap, initial gap	Max. suffix, prefix length	Affix classifiers	Affix memo-rization length	ngram model pa-rameters
Arabic	5, 2	4, 4	Suffix	0	6, normalized
	5, 2	4, 4	Suffix, prefix	0	6, unnormalized
Finnish	1, 0	5, 0	Suffix	3	6, normalized
	1, 0	6, 0	Suffix	3	6, normalized
Georgian	5, 1	5, 3	Suffix	0	5, unnormalized
	5 ,1	5, 3	Suffix	0	5, unnormalized
German	2, 2	5, 3	Suffix	3	6, normalized
	2, 2	5, 3	Suffix	3	6, normalized
Hungarian	1, 0	5, 0	Suffix	3	6, normalized
	1, 0	6, 0	Suffix	3	5, normalized
Navajo	3, 7	5, 3	Prefix	3	6, unnormalized
	3, 7	6, 3	Prefix	0	6, unnormalized
Russian	1, 3	5, 0	Suffix	0	3, normalized
	1, 3	6, 3	Suffix	3	6, normalized
Spanish	2, 2	5, 0	Suffix	3	6, normalized
	5, 2	6, 3	Suffix	3	6, normalized
Turkish	1, 0	5, 0	Suffix	3	5, unnormalized
	5, 2	6, 0	Suffix	3	6, normalized

Table 5: System parameters for inflection (upper) and lemmatization (lower) tasks.