Developing a Morphological Segmenter for Russian

America L. Holloway
Swarthmore College
Swarthmore, PA 19081
ahollow1@swarthmore.edu

Abstract

This paper presents an algorithm for developing a morphological segmenter for Russian. The segmenter can find multiple prefixes and suffixes for any given word. Therefore it is more suitable for a highly inflected language than a segmenter that is limited to at most one prefix or suffix. The segmenter requires a small hand segmented corpus to bootstrap from, and a larger unsegmented corpus from which to learn. The algorithm uses trigram probabilities, and Witten-Bell smoothing to predict the correct segmentation of a word. A filtering step is also used to weed out bad segmentations.

1 Introduction

Many languages, including Russian and Arabic, have a richer morphology than is found in English. In Russian, not only do verb endings change to reflect person, gender and number, noun endings also change (or in some cases are truncated) to reflect case. For example, the ending а is appended to a masculine noun to form the genitive singular. Furthermore, a word in Russian can often times be decomposed into smaller units, or morphemes, each of which carries its own meaning. These morphemes contribute to, and refine, the meaning of the entire word. For example, the noun председатель (*predsedatil*) means 'representative'. Literally, it can be translated as 'the one' (ель) 'who sits' (сидеть) 'before' (пред), or more figuratively, 'the one who represents' us. In general, it is not rare for a word to have multiple prefixes and suffixes. Multiple suffixes, in particular, are common. As an example, reflexive verbs will always have two suffixes. The first suffix -ся (*-cya*) indicates it is a reflexive verb, and the second suffix indicates what type of verb it is. These suffixes include -оваь (*-ova*), -ать(*-at*) and -ить (*-ut*). Thus, to capture the morphology of such a language, it is important that any morphological analyzer be able to recognize multiple prefixes and suffixes.

The algorithm presented in this paper is adapted from the morphological segmenter for Arabic created by (Lee et al., 2003). Many existing morphological analyzers, for example (Goldsmith, 2000), identify only single suffixes. This type of system fails to capture the entire morphology of Russian. Recognizing multiple prefixes and suffixes is especially important for tasks such as aligning corpora, information retrieval and machine translation. This is because one Russian word may correspond to multiple words in a different language. Thus, our goal was to implement a segmenter that (1) could identify multiple affixes and (2) required few resources, in order to create a superior morphological analyzer specifically for Russian.

Our system requires only a small hand-segmented corpus to bootstrap the segmenter, and a larger, unsegmented corpus from which to gain new stems. For any Russian word, all possible segmentations are enumerated and the trigram probability of each is computed. The highest scoring segmentation is chosen as the correct one. The system performance is

Appeared in: *Proceedings of the Class of 2005 Senior Conference*, pages 28–33
Computer Science Department, Swarthmore College

surprisingly good given the small corpus and simple algorithm.

2 Related Work

As stated, this algorithm draws heavily from (Lee et al., 2003). They present a morphological segmenter for Arabic which identifies multiple prefixes and suffixes and requires only a small hand segmented corpus (110,000 words) and a large unsegmented corpus (155 million words). They also supplement their segmenter with an additional prefix/suffix list. The large unsegmented corpus is used to acquire new stems. They first divide the corpus into partitions. For each word, all possible segmentations are enumerated and the segmentations with the highest probabilities are kept. After each partition, the trigram probabilities are recomputed to take into account the new stems found. Each stem is also subjected to further testing to ensure that it does not contain a prefix or suffix. Stems are added to the list based upon the stem frequency (i.e. the number of times they are seen), the probability that a substring of the stem is a prefix or suffix, and contextual information. With just trigram probabilities alone, (Lee et al., 2003) are able to reduce the error from the baseline performance by half.

(Goldsmith, 2000) uses the notion of minimum description length (MDL) to implement a morphological segmenter, *Linguistica*, that is quite successful. *Linguistica* takes only a corpus and returns a list of stems, a list of suffixes and a list of signatures. A signature is a set of suffixes which can appear on the end of a stem. An example signature is the set (-NULL, -s). There are many stems, such as *apple* or *cow*, that are associated with this signature. A first analysis of the corpus can be as simple as splitting every word after each letter. Other heuristics are then employed to shrink the list of signatures.

Minimum description length is based on the notion that the number of letters in the morphological analysis of a corpus (e.g. a list of stems, suffixes and signatures) will be less than the number of letters in the original corpus. Accordingly, Goldsmith develops a description length to measure the size of the morphological analysis of a corpus. That is, he creates a description length to measure the size of the stem list, suffix list and signature list. After each heuristic is applied, the description length is computed. If the description length has decreased, the analysis is kept. Notably, *Linguistica* identifies only one suffix per word. For example, if the word *breathings* occurred in our corpus, the stem would be *breathing* and the suffix would be *-s* [1]. Thus *breathings* would be associated with the signature given above. Recall of 85.9% and precision of 90.4% is achieved for English.

Work using multilingual corpora to aid in morphological analysis has also been performed. (Yarowsky et al., 2001) use a lemmatizer and multilingual corpora to achieve a precision over 98% on a French corpus of 1.2 million words. (Hana et al., 2004) use a Czech-Russian aligned corpus. The system combines information from their own morphological segmenter, the Czech corpus and a part of speech tagger. Instead of detecting multiple prefixes or suffixes, they use the notion of paradigms. A paradigm is a list of suffixes, along with the corresponding part of speech, that can be appended to a certain class of stems. One interesting technique used to find the correct suffix of a word is the longest-suffix approach. Simply put, the correct suffix is usually the longest one. We have adopted this heuristic to increase our system performance.

3 Morphological Segmenter

3.1 Parsing Words

Before discussing the algorithm used to build the morphological segmenter, it is important to discuss what constitutes a prefix or a suffix. Two categories of suffixes are distinguished by the segmenter: suffixes that change the part of speech, and suffixes that preserve part of speech, but reflect a change in case, or person.

As an example of the first type of suffix, consider the suffix -ение (*-enie*). This is appended onto the end of a verb to form the corresponding noun. Hence, the noun обсуждение (meaning 'discussion') is derived from the verb обсуждать (meaning 'to discuss'). To then form the genitive or possessive form of the noun, the ending -ие (-*iye*) changes to -ия (-*iya*). This is an example of the second type of suffix which preserves the part of speech.

[1] This example is taken from (Goldsmith, 2000)

Appeared in: *Proceedings of the Class of 2005 Senior Conference*, pages 28–33
Computer Science Department, Swarthmore College

count			prefix	stem	suffix(es)
7	&	&	#N	угл	+ов
16	&	&	#N	каз	2+ан +ие
17	&	&	#N	завод	+N
23	&	&	#раз	дел	+ить
29	&	&	#бес	платн	+ые
35	&	&	#N	север	+е
14	&	&	#N	тысяч	+ами

Table 1: Morphologically Segmented List of Russian Words

In general, noun or adjective prefixes are harder to discern than verbal prefixes or suffixes. A verbal prefix is often used to denote aspect. However with nouns (and adjectives) a prefix neither changes the part of speech, nor the case, person or number. Instead we chose to define a prefix as a morpheme that refines or adds to the meaning of the word. For example, appending the preposition без (meaning 'without' or 'short of') to the adjective умный (meaning 'of the mind') gives the adjective безумный which means 'crazy'. In general however, the presence of a preposition at the beginning of a word does not necessarily mean it is acting as a prefix. Thus our method of determining prefixes for nouns and adjectives is inherently subjective. To account for this, when creating the small hand segmented corpus, a verb was determined to have a prefix if it was shown to have one in the Oxford Russian Dictionary. Nouns were determined to have a prefix, again, if a prefix was shown in the Oxford Russian Dictionary, or if it was clear from the meaning. The subjective nature of determining whether or not a noun contains a prefix actually hurt the performance of the segmenter and is discussed in the Results section.

3.2 Bootstrapping

A small hand segmented corpus of 474 Russian words was used to bootstrap the segmenter. Each word was split into prefix(es), stem and suffix(es). We adopt the convention that a pound sign (#) precedes every prefix, and a plus sign (+) precedes every suffix. In order for every word to have at least one prefix and suffix, the letter N is used for the null prefix and suffix. Finally, for the purpose of calculating trigram probabilities, two symbols (& &) were placed at the beginning of each word. Table 1 shows a sample of the corpus used to bootstrap the segmenter. From the corpus we create a static list of suffixes and prefixes, and a list of stems to which we will be adding. The smaller corpus is also used for initial trigram probabilities.

prefix	stem	suffix
N	заработк	е
N	заработке	N
за	работк	е
за	работке	N

Table 2: All possible suffix-prefix segmentations

3.3 Building the Segmenter

The larger corpus consists of approximately 40,297 words and is split into 403 partitions of 100 words each. The number of words in the partition was arbitrarily chosen. We first read in an entire partition. Then for each word w, all possible segmentations of w are enumerated, and the probability for each segmentation is calculated. Only the segmentation with the highest probability is kept. The stem is then added to a list of possible stems. When the frequency (i.e. the number of times the stem has been seen) passes a given threshold, the stem is added to the list of accepted stems. Since the larger corpus is relatively small, the threshold value was set at 2.

3.3.1 Segmenting Words

Given any Russian word w, we wish to find all possible prefixes and suffixes of w. To find all possible prefixes of a given Russian word w, we compare substrings of w against the list of prefixes. The first substring is simply the first letter of w. The next substring is the first two letters of w, then the first three, and so on, until we come to the end of the word. We do the same for suffixes except we begin at the last letter of w. We then enumerate all possible prefix-suffix combinations. The null prefix (suffix) is always a possible prefix (suffix) for every word. Table 2 shows all the prefix-suffix combinations for the word заработке (*zarabotke*), the prepositional form of the word заработок meaning 'earnings'.

Appeared in: *Proceedings of the Class of 2005 Senior Conference*, pages 28–33
Computer Science Department, Swarthmore College

3.3.2 Filtering

Often longer suffixes include within them shorter suffixes. For example, the word живому (*zhivomy*) has two possible suffixes: -ому or -у. In general however, the longest suffix is usually the correct one. A suffix on the end of a word of length 5 is more likely to be the correct one, than a suffix that is only of length 1. Thus, we give preference to longer suffixes. If a word has one (or more) compound suffixes, we consider only the compound suffixes and disregard any other possible segmentation of the word with only one suffix (including the null suffix).

We also provide to the system a list of 8 default suffixes. If a word contains one of these suffixes, all other segmentations of the word are disregarded except for this one. Hence there will be only one segmentation for the word, the segmentation with the default suffix.

In Russian, certain word endings will almost always indicate a suffix. For example, the genitive ending for masculine adjectives is ого (*-ovo*). An adjective will never have this ending unless it is in genitive case, and very few nouns have this ending. So few, that it is worth making ого a default suffix.

3.4 Probabilities

Given any Russian word w and any possible segmentation of w into morphemes $m_1 m_2 m_3 ... m_k$, the probability of the segmentation is given as:

$$P(\&) * P(\&|\&) * P(m_1|\&\&) * ... * P(m_k|\&\&m_1...m_{k-1}) \quad (1)$$

We can simplify this expression using a second-order Markov assumption. This makes computing the probability of morpheme m_i easier, since the probability of seeing m_i can be estimated given the previous two morphemes instead of all preceding morphemes. Also, Since every word begins with &&, we can consider $P(\&)$ and $P(\&|\&)$ to be constants and thus disregard them. This gives

$$P(m_1|\&\&) * P(m_2|\&m_1) * ... * P(m_k|m_{k-2}m_{k-1}) \quad (2)$$

We use the maximum likelihood estimate (MLE) shown below to calculate $P(m_i|m_{i-2}m_{i-1})$.

$$P(m_i|m_{i-2}m_{i-1}) = \frac{C(m_{i-2}m_{i-1}m_i)}{C(m_{i-2}m_{i-1})} \quad (3)$$

Witten-Bell discounting (Witten and Bell, 1991) is used for smoothing. The probability of seeing $m_{i-2}m_{i-1}m_i$ for the first time can be approximated by the number of times we saw previous trigrams for the first time. Let $m_{i-2}m_{i-1}m_i$ be a trigram that has never before been seen. Then $P(m_i|m_{i-2}m_{i-1})$ can be expressed as:

$$P(m_i|m_{i-2}m_{i-1}) = \frac{T}{Z(N+T)} \quad (4)$$

where T is the number of unique trigrams observed before, N is the total number of trigrams seen before, and Z is the number of zero trigrams. The probability of seeing $m_{i-2}m_{i-1}m_i$ is given by the number of previous times we saw a trigram for the first time (T) divided by the number of times a new trigram could have have been seen for the first time (N+T). We then distribute this probability evenly to all of the zero trigrams by dividing by Z. Since we need to know the value of Z in advance, we must read an entire partition first, segment all the words, and keep track of how many segmentations result in a stem that has never before been seen.

4 Results

To evaluate the segmenter, the hand tagged corpus was split into 9 different sets. Each set contains a different 50 lines from the corpus to test on, and the remaining 428 lines from which to train. Thus, the first set used the first 50 lines from which to test, the second set used the second 50 lines from which to test, and so on. The last set, set 9, was tested on the last 77 lines.

The segmenter was trained on the hand tagged corpus, and then asked to segment the appropriate 50 lines. The segmenter was evaluated according to recall and precision. Table 3 shows the performance of the segmenter on sets 1 through 9. The first column shows the recall of the segmenter (i.e. of the correct prefixes and suffixes, how many did the segmenter find). The second column shows the precision (i.e. of the prefixes and suffixes postulated by the segmenter, which were correct). The third and

Appeared in: *Proceedings of the Class of 2005 Senior Conference*, pages 28–33
Computer Science Department, Swarthmore College

	including null		excluding null	
Test Set	Recall	Precision	Recall	Precision
1	77.6%	81.90 %	75.6%	76.92%
2	90.52%	92.85 %	81.25%	78.94%
3	78.26%	83.33 %	64.29%	78.95%
4	74.55%	75.93 %	62.69%	72.73%
5	81.65%	86.53 %	87.93%	79.17%
6	81.65%	86.53 %	87.93%	79.17%
7	82.20%	86.61 %	68.57%	81.67%
8	82.46%	84.55 %	78.13%	77.19%
9	85.45%	83.03 %	77.42%	81.18%
Average	**88.74%**	**84.58 %**	**75.98 %**	**78.43 %**

Table 3: The first two columns show recall and precision when the null prefix/suffix is included. The last two columns show recall and precision disregarding the null prefix/suffix

fourth column show the recall and precision without taking the null prefix and suffix into account.

4.1 Discussion of Errors

Given the small size of the training corpora, and the simple nature of the algorithm, the results are encouraging. A majority of the mis-segmentations stem from a few key errors. One of the biggest problems was the small size of the hand tagged corpus. A few stems were seen once or twice and hence the corresponding suffix had an extremely high probability. For example the suffix -ь was seen only once with the word лечь. Since the probability of a segmentation was determined using trigram counts (Equation 3), the probability of the suffix -ь was 1.

One disheartening result is that the segmenter failed to find any prefix save one. However, since there were so few prefixes in the hand-tagged corpus, performance was not hurt too drastically. The poor prefix performance can be attributed to the subjective nature of prefixes. A word was considered to have a prefix if (1) it was shown with a prefix in the dictionary, or (2) the prefix contributed to the meaning of the word and taking away the prefix gave another related word. Thus, two words w_1 and w_2 may both have the same first two letters, yet only w_1 has a prefix. This, combined with the small corpus size and overwhelming probability of the null prefix, accounts for the system's preference for the null prefix.

5 Conclusion and Future Work

The segmenter does surprisingly well taking into account the small corpora size and the rather simple algorithm. In general, it is easy to detect a majority of suffixes, either because they are very unique, or because they are rather long. It is a small subset of suffixes such as -о, -е and -а that are difficult to identify. Thus, focusing on identifying these suffixes would result in major system gain. Another area of interest is a more uniform way of segmenting words into prefix(es), stem and suffix(es). In particular, changing the method of prefix identification so that every word with a particular first few letters are considered to have the same prefix, even if this prefix does not contribute to the meaning of the word.

6 Acknowledgments

We would like to thank Nastassia Herasimovich for helping segment Russian words and Professor Wicentowski for all of his (very much needed) help.

References

Goldsmith J. 2000. *Unsupervised learning of the morphology of a natural language* Computational Linguistics, 27(1).

Hana J., Feldman A. and Brew C. 2004. *A Resource-light approach to Russian morphology: Tagging Russian using Czech resources.* Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing.

Appeared in: *Proceedings of the Class of 2005 Senior Conference*, pages 28–33
Computer Science Department, Swarthmore College

Witten I.H. and Bell T.C. 1991. *"The Zero-Freqency Problem: Estimating the probabilities of novel events in adaptive text compression"* IEEE Transactions on Information Theory, 37(4), p. 1085-1094.

Yarowsky D., Ngai G. and Wicentowski R. 2001. *Inducing Multilingual Text Analysis Tools via Robust Projection across Aligned Corpora.* Proceedings of the HLT 2001, pages 161-168.

Lee Y., Papineni K. and Roukos S. 2003. *Language Model Based Arabic Word Segmentation.* Proceedings of the 41st Annual Meeting of the Association, pages 399-406.

Appeared in: *Proceedings of the Class of 2005 Senior Conference*, pages 28–33
Computer Science Department, Swarthmore College