COMPARISON OF LANGUAGE MODELLING TECHNIQUES FOR RUSSIAN AND ENGLISH

E.W.D.Whittaker *P.C.Woodland*

Cambridge University Engineering Department,
Trumpington Street, Cambridge CB2 1PZ, UK.

{ewdw2,pcw}@eng.cam.ac.uk

ABSTRACT

In this paper the main differences between language modelling of Russian and English are examined. A Russian corpus and a comparable English corpus are described. The effects of high inflectionality in Russian and the relationship between the out-of-vocabulary rate and vocabulary size are investigated. Standard word and class N-gram language modelling techniques are applied to the two corpora and perplexity results are reported. A novel approach to the modelling of inflected languages is proposed and its efficacy compared with the other techniques.

1. INTRODUCTION

Much work has been conducted in recent years on language modelling techniques for speech recognition of English. In contrast, less commercially attractive yet widely spoken languages like Russian have received comparatively little attention in the literature (the first reported large-vocabulary recogniser for Russian appeared only recently[3]). Moreover, there are important difficulties with modelling Russian which are also present in many other languages. In this paper, we take two well-known statistical language modelling techniques which work well for English and compare their performance for modelling Russian.

Russian differs from English in two important ways when considering statistical language modelling. Firstly, Russian is a highly-inflected language—almost all content words have several inflections (word-endings) which change the grammatical case, gender, number, etc. of the word—and a vocabulary of comparable utility to that for English requires that the number of vocabulary words be an order of magnitude greater. Secondly, a consequence of this inflectionality is a relaxation in the word order of Russian words. In practice however, a completely free ordering of words is not observed, and regular stylistic patterns are seen. Changes in word order generally serve to lend more weight to particular words in the sentence. In this paper, the salient characteristics of a Russian corpus that has been compiled, will be compared against a well-known English corpus.

Many Russian words are formed in a "composite" fashion. The addition of affixes to a root "string" forms a word-stem which generally has a new lexical meaning to that of its root. The affixes which are used as building blocks in this way are common to many words, and often colour the new words in a predictable way. Particle (sub-word) N-gram language models will be built for Russian using a fixed set of particles. Preliminary results for models with arbitrary and optimised decompositions of words into these particles will be presented.

2. CORPORA CHARACTERISTICS

2.1. The Russian Corpus

The Russian corpus is a varied collection of around 2,500 literary and non-literary texts, covering several genres and styles of composition. After the automated editing of common typographical errors, the regularisation of formatting and the insertion of sentence boundary markers, the corpus contained approximately 100 million words. The final corpus was then partitioned into training, development (dev-test) and evaluation (eval-test) sets, in the ratio 98:1:1. The important characteristics of the partitions are given in Table 1.

	train	dev-test	eval-test
Total words	101,819,592	1,036,719	1,040,173
Unique words	1,018,856	115,472	116,029

Table 1: Russian corpus partitions

2.2. The English Corpus

The British National Corpus (BNC) was chosen for use in the English-language experiments, due to its similarity to the Russian corpus in terms of composition and size. After editing to correct common inconsistencies in the corpus, partitions for training, dev-test and eval-test were obtained in a similar ratio to that for the Russian corpus. The characteristics of the resulting partitions are given in Table 2

	train	dev-test	eval-test
Total words	113,522,206	1,018,958	998,680
Unique words	406,653	37,072	37,960

Table 2: BNC corpus partitions

2.3. Effect of vocabulary size on OOV-rate

The graph in Figure 1 shows how the percentage of out-of-vocabulary (OOV) words—words occurring in the test set which did not occur during training—varies with the size of the vocabulary for both corpora. The rate at which the percentage of OOV words decreases for the BNC is much greater than for the Russian corpus. From Figure 1, it is seen that the vocabulary for Russian

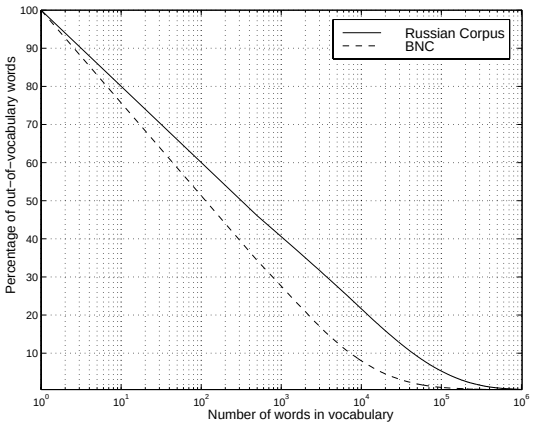

Figure 1: Variation of OOV-rate against (log) vocabulary size

must, on average, be an order of magnitude larger than for the BNC, in order to obtain a similar OOV-rate. For example, a 65k vocabulary has a 1.2% OOV-rate on the BNC corpus compared to 7.5% on the Russian corpus. The Russian corpus would require a vocabulary of 375k words in order to achieve a 1.2% OOV-rate. A possible solution to reducing the OOV-rate is to decompose words into smaller lexical units.

2.4. Word N-gram language model

For both corpora, a baseline word backoff trigram language model employing Good-Turing discounting was built, using the CMU-Cambridge Toolkit [1]. For each corpus a vocabulary of the most frequent 65k words in the training set was used. In Table 3, perplexity figures are given for two trigram models built on both corpora. One model has the bigram and trigram cutoffs set to one, and the other has the cutoffs set to zero. Hit-rates for the corresponding word trigram models are also given. N-gram hit-rates express the percentage of 1-grams, 2-grams, etc. used in computing the perplexity. The frequency-of-frequency statistics for the two corpora are remarkably similar, yet it is instructive to note that when all singleton N-grams are retained by the models, the perplexity on the Russian test data is reduced by 12.9% compared to only 1.7% for the BNC. Retaining N-gram singletons results in a four-fold increase in the model size for both languages, so for the following experiments the baseline models do not include them.

The perplexities for the Russian corpus are notably higher than for the BNC. There are two apparent explanations for this; the first is related to data sparsity in that the high percentage of OOV words for the 65k vocabulary results in less specific histories. In the Russian corpus, 15% of trigrams which were "hit" had one or

	Russian Corpus		BNC	
Cutoffs (bi,tri)	1, 1	0, 0	1, 1	0, 0
Perplexity	407.2	354.7	240.8	236.8
3-gram hits (%)	54.7	63.6	60.3	67.1
2-gram hits (%)	31.6	26.7	30.2	25.9
1-gram hits (%)	13.7	9.7	9.5	7.0

Table 3: Perplexity figures and N-gram hit-rates for the two corpora's word trigram models with bigram (bi) and trigram (tri) cutoffs both set to one, and both set to zero

more unknown words in the history. This compares to only 2.5% for the BNC. The second potential explanation is that N-gram models are not suited to the relaxed word-ordering which occurs in Russian. The trigram hit-rates in Table 3 appear not to support this argument. However, the significant perplexity reduction when singletons are retained and the high proportion of unknown words in the histories suggest otherwise.

3. CLASS N-GRAM MODELS

3.1. Overview

The effects of sparsity in a corpus can be partially overcome by mapping the words, w, of the vocabulary, V, into classes, C—where $|C| < |V|$—and then collecting N-gram statistics for the mapped corpus. A deterministic word-to-class mapping,

$$C : w \rightarrow c = C(w), \tag{1}$$

in which a word may only belong to one class, can be obtained using an automatic clustering algorithm—words are grouped into classes based on some similarity criterion. The algorithm used in these experiments is based on Neys's method [4]. Only one iteration through the vocabulary is performed, to allow a fair comparison of all the class models.

The class-based model generates a probability for a word as a product of the probability of the word's membership of its class and the probability of that class given a history of classes, as for the following bigram class model:

$$P(w_n \mid w_{n-1}) = P(w_n \mid C(w_n))P(C(w_n) \mid C(w_{n-1})). \tag{2}$$

Models can be interpolated to combine the generality of the class model with the more specific nature of the word model, for example.

3.2. Results

Clustering was performed with 204, 504, 1004 and 2004 classes. Four words (sentence boundary markers etc.) were not clustered and each remained in its own class; other words could not be moved to these classes. Perplexity figures for the class N-gram models and the model which results from interpolating the class-based and word trigram models are given in Table 4 for the Russian corpus and Table 5 for the BNC. The interpolation weights were chosen by optimising their values using the appropriate dev-test set for each corpus. The final column in each table indicates

the relative improvement of the interpolated model over the baseline word trigram model. Some examples of the classes obtained are given in Tables 6 and 7.

No. of classes	Perplexity		interp. wgts(wd,cl)	rel to word model (%)
	class-based	interp.		
204	784.4	377.9	0.73, 0.27	7.2
504	586.0	359.3	0.64, 0.36	11.8
1004	490.4	347.5	0.56, 0.44	14.7
2004	440.0	343.4	0.53, 0.47	15.7
65000	407.2	—	—	—

Table 4: Perplexities on Russian corpus for class trigram models and interpolated word/class models for eval-test data

No. of classes	Perplexity		interp. wgts(wd,cl)	rel to word model(%)
	class-based	interp.		
204	398.3	227.4	0.74, 0.26	5.6
504	322.5	222.4	0.66, 0.34	7.6
1004	284.7	220.1	0.60, 0.40	8.6
2004	261.2	220.6	0.53, 0.47	8.4
65000	240.8	—	—	—

Table 5: Perplexities on BNC for class trigram models and interpolated word/class models for eval-test data

bez *(without Prep.)*, lishjenii *(lacking MscNomSg)*, lishionaia *(lacking FemNomSg)*, lishjenie *(lacking PlNomSng)*. . .
khotel *(wanted MscSg)*, zahotel *(had wanted MscSg)* zhelal *(desired MscSg)*, predpochiol *(preferred MscSg)*. . .
moevo *(mine Gen)*, nashevo *(ours Gen)*, vashevo *(yours PlGen)* tvoevo *(yours SgGen)*, ch'evo *(whose SgGen)*. . .

Table 6: Three examples of classes of Russian words with their meanings and part-of-speech. *Prep*=preposition, *Msc*=masculine, *Fem*=feminine, *Nom*=nominative, *Gen*=genitive, *Sg*=singular, *Pl*=plural

3.3. Discussion

As can be seen from Tables 4 and 5, similar relative improvements are obtained for both corpora when the class-based models are interpolated with the word model. It is interesting to note that better absolute improvements in perplexity are obtained for the Russian corpus and that the optimal number of classes is greater for Russian. Closer examination of the contents of classes obtained for Russian reveals many clear groupings: e.g. semantically similar words with a particular grammatical inflection; and, all inflected forms of one word. It is much easier to attach a consistent linguistic meaning to the contents of the Russian classes than to the contents of the English classes. It should be noted that better absolute results can be expected across all models by allowing further iterations during clustering, and by using longer N-grams in the class models.

whence, where, wherein, whereof, whither. . .
brother's, companion's, daughter's, father's, grandfather's, grandmother's, grandparents', husbands'. . .
better, braver, mightier, nicer, truer. . .

Table 7: Three examples of classes of English words

4. PARTICLE N-GRAM MODELS

In a similar vein to the methods described in [2], the first experiments we performed with particle (sub-word) N-gram models used a data-driven method to isolate useful particles for modelling vocabulary words. However, a satisfactory method for decomposing words was not found and so the preliminary work described here concentrates on optimising the decompositions of words using a set of particles that was defined beforehand. Preliminary results will be given for particle models using an arbitrary decomposition of words and an optimised decomposition of words.

4.1. Motivation

Russian words often exhibit clearer morphological patterns than can be found in English words. If we examine a simplified model of a Russian verb [5], we can determine the presence of several constituent parts: a *root* which can be thought of as responsible for the nuclear meaning of the verb, attached to which may be zero or more *derivational prefix(es)* and zero or one *suffix*, which together form a *stem*. The *stem* often acquires an entirely new lexical meaning with the presence of these affixes. An *inflection* which is appended to the stem determines the grammatical case, gender, number etc. of the *word*. All the points in the word where the constituent parts are joined can be considered *morpheme boundaries*. Obviously, this is an idealised example, although the "synthetic" nature of Russian is also clearly visible in Russian nouns, adjectives and participles.

Given an initial decomposition for every word, the optimisation algorithm described below attempts to determine the best decomposition of words, $U(w)$, into a fixed set, Ψ, of particles u_i,

$$U : w \rightarrow U(w) = u_0, u_1, \ldots, u_A \quad u_i \in \Psi.$$

Only the orthography of words and the statistics of their occurrence in the corpus are used by the algorithm. Since the identity of a morpheme is distorted by the environment in which it occurs, we can expect that the decompositions which are obtained will not necessarily correspond to conventional linguistic decompositions of words into morphemes. Indeed, a decomposition of words into uninflected stems plus inflections, rather than into constituent *morphs*, may be preferred but since the algorithm does not assume any prior linguistic knowledge about the language, we have little control over which decompositions are obtained.

Given a decomposition for each word, the conditional word probability can be computed, for example, with a particle bigram model as follows:

$$\begin{aligned} P(w_n \mid w_{n-1}) &= P(u_0^{w_n}, \ldots, u_A^{w_n} \mid u_0^{w_{n-1}}, \ldots, u_B^{w_{n-1}}) \\ &= P(u_A^{w_n} \mid u_{A-1}^{w_n}) \cdot P(u_{A-1}^{w_n} \mid u_{A-2}^{w_n}) \cdots P(u_0^{w_n} \mid u_B^{w_{n-1}}) \cdot \\ &\quad P(u_B^{w_{n-1}} \mid u_{B-1}^{w_{n-1}}) \cdots P(u_1^{w_{n-1}} \mid u_0^{w_{n-1}}). \end{aligned} \quad (3)$$

Relative frequencies of the occurrences of pairs of particles can be used to compute the above conditional probabilities and smoothed in the same way as for conditional word probabilities.

4.2. Word decomposition algorithm

The word decomposition algorithm seeks to optimise the decomposition of each word in an iterative reestimation and maximisation fashion. For each word in turn, the probability of the decomposition of a word into particles is maximised, and the particle statistics are reestimated. This process of local maximisations is repeated for every word, until no further improvement is obtained.

The fixed set of particles, Ψ, is chosen to be all possible single characters, and (up to) the 1000 most frequent n-tuples ($n = 2, 3 \ldots$) of characters, occurring in vocabulary words in the corpus. The initial decomposition is generated by splitting words into this set of particles by finding the largest particle that occurs at the beginning of the word, then finding the largest particle in the remainder of the word, and so on until the end of the word is reached. Unigram and bigram particle statistics are collected from these initial word decompositions, including particle contexts which cross word boundaries. For each word, w, in turn, the best path through w is found, and the particle statistics for this new path are updated. The best path is the path with the highest probability which is computed as the product of conditional particle bigram probabilities for particles within w, multiplied by the sum of weighted cross-word particle probabilities for all words that co-occur with w. The weight for each cross-word probability is the bigram count for the two words divided by w's unigram count.

4.3. Results

The vocabulary to be decomposed was the same 65k words used in the Russian word model. The most frequently occurring 1000 words and words shorter than 4 characters long were not considered for decomposition. Starting with the initial decomposition of words, the algorithm performed three iterations over all vocabulary words until there was no further change in any word's decomposition. In total, about 950 word decompositions were improved. On average, there are 2.9 particles per vocabulary word.

Standard backoff particle trigram and 4-gram models were built using the initial decompositions of words, and using the decompositions after reestimation by the algorithm. There were 5340 particles in the model built using the initial word decompositions (init), and 5334 in the model using the optimised decompositions (final). The N-gram ($N = 2, 3, 4$) cutoffs for all models were set to one. The final 4-gram model was 0.1% smaller than the init 4-gram model. Word level perplexities are given in Table 8 along with perplexities for the model formed by the interpolation of the particle model with the word model, and with the 2004-class model from Section 3.

4.4. Discussion

From Table 8 we observe that small improvements in perplexity are obtained when the particle model is interpolated with the word-based model and also with the class-based model. The fact that some improvement is obtained, even with the particle trigram model, suggests that the method is tackling the data sparsity problem to some extent. The optimisation algorithm has not improved the perplexity on the test data, but did tidy up several initial decompositions into more intuitive decompositions. The initial set of particles was not an optimal choice and a better approach may be to select initial particles according to their usefulness. In addition, the final decompositions are highly dependent on the initialisation that is used, and some means of cross-validation or perturbing the decompositions may find more optimal decompositions.

Model type	Perplexity		interp. wgts (wd, pt, cl)	rel to word model(%)
	eval-test	interp.		
init 3g	585.7	396.1	0.78, 0.22, 0	2.7
init 4g	455.3	381.8	0.58, 0.42, 0	6.3
final 3g	586.0	396.1	0.78, 0.22, 0	2.7
final 4g	455.0	381.6	0.58, 0.42, 0	6.3
final 4g	455.0	333.8	0.34,0.31,0.35	18.0

Table 8: Perplexities on Russian corpus for particle(pt) 3- and 4-gram models before (init) and after (final) optimisation, and interpolated particle, word(wd) and 2004-class(cl) trigram models

5. CONCLUSION

In this paper, we have highlighted the different characteristics of Russian and English and discussed methods to alleviate the acute data sparsity effects caused by the vastly increased vocabulary size. Class models have been built for the Russian corpus and shown to provide better relative improvements in perplexity than were obtained for English. N-gram particle models have been built for Russian and shown to improve the perplexity when interpolated with word- and class-based models. A statistical method for optimising the decomposition of words into particles has also been presented but not shown to improve the perplexity. Further work on the particle model will concentrate on the groupings determined by the clustering algorithm which had the unexpected property of grouping together words with similar morphological patterns. Also, the technique for optimising word decompositions will be incorporated into a particle selection mechanism.

6. REFERENCES

1. P. R. Clarkson and R. Rosenfeld. Statistical Language Modeling Using the CMU-Cambridge Toolkit. In *Proceedings Eurospeech*, 1997.
2. N. D. Andreeva (editor). *Statistiko-Kombinatornoe Modelirovanie Iazykov*. Nauka. Moscow, Leningrad, 1965. (In Russian).
3. D. Kanevsky, M. Monkowski, and J. Sedivy. Large vocabulary speaker-independent continuous speech recognition in Russian language. *SPECOM96*, 1996.
4. H. Ney, U. Essen, and R. Kneser. On structuring probabilistic dependences in stochastic language modelling. *Computer Speech and Language*, 8:1–38, 1994.
5. J. Caflisch. Sr. *Issues in Russian Linguistics*. University Press of America, Inc., 1995.