Using Part of Speech N-Grams for Improving Automatic Speech Recognition of Polish

Aleksander Pohl[1,2] and Bartosz Ziółko[1]

[1] Department of Electronics
AGH University of Science and Technology
al. Mickiewicza 30, Kraków, Poland
www.dsp.agh.edu.pl
[2] Department of Computational Linguistics
Jagiellonian University
ul. Łojasiewicza 4, 30-348 Kraków, Poland
www.klk.uj.edu.pl
aleksander.pohl@uj.edu.pl, bziolko@agh.edu.pl

Abstract. This paper investigates the usefulness of a part of speech language model on the task of automatic speech recognition. The develped model uses part of speech tags as categories in a category-based language model. The constructed model is used to re-score the hypotheses generated by the HTK acoustic module. The probability of a given sequence of words is estimated using n-grams with Witten-Bell backoff.

The experiments presented in this paper were carried out for Polish. The best obtained results show that the part-of-speech-only language model trained on a 1-million manually tagged corpus reduces the word error rate by more than 10 percentage points.

1 Introduction

In the most of modern automatic speech recognition (ASR) systems the algorithm of operation is as follows: the input signal is recognized using an acoustic model (AM), i.e. a model that describes the relation between the sounds and the phonemes from the chosen alphabet. In that way a language message hypothesis is encoded using the sound. Since, typically that model is not analyzing all possible data, the result of the AM module is not a single result, but a list of hypotheses with probability estimations or a lattice. Then, a language model (LM), i.e. a model that describes the relations between the words is used to rescore the acoustic hypotheses in order to select the hypothesis that is the most plausible according to the LM and AM.

Since a LM utilizing all the various relations between the words is very hard to built, there are many approximations, word n-grams being the most popular [1]. In the word n-gram LM the relations between the words are narrowed down to the order in which the words appear. It is believed that the probability of the next word in a sequence can be reasonably estimated, given the n-1 preceding words. Given a large corpus it is possible to estimate these probabilities and

P. Perner (Ed.): MLDM 2013, LNAI 7988, pp. 492–504, 2013.
© Springer-Verlag Berlin Heidelberg 2013

use them to compute a probability of any given sequence of words. However, it should be mentioned, that the efficiency of an n-gram LM is language dependent. English, being the most important and common language for speech recognition research, caused popularity of n-grams. But, it has to be stressed, that English is positional, and as such, n-grams are very natural to be used and close to the logics of the language. In case of Polish and other inflectional languages like Finnish, Turkish and other (than Polish) Slavic languages, the order of the words is not the main logics of the language. As a result, an n-gram LM taken directly from ASR system designed for English is not necessarily the best choice.

For instance in Polish the expressions *dom Adama* and *Adama dom* (*Adam's house*) although not equally probable, express the same relation between these words. What is more the number of tokens in Polish and other inflectional languages is larger than in English, since words have many inflectional forms (e.g. *Adam, Adama, Adamowi, Adamem, Adamie, Adamowie, ...* are all forms of *Adam*).

The primary problem connected with n-gram based LMs is data sparsity – it is impossible to collect a corpus that would allow to compute the probabilities for any word sequence of a given length that might appear in the recognized speech. As a result there are sequences that lack probability estimation. Due to the fact that the number of tokens in inflectional languages is larger than in positional languages, this problem is amplified in the case of the former. Partial solution to this problem is usage of techniques such as smoothing, interpolation and backoff [2].

The second important problems are parentheses which seriously both introduce errors to LM if they appear in the training corpus, and can be very difficult to be recognized if they appear in speech.

There are also approaches that do not assume that words are the best building blocks of the LM. There are models based on morphs, morphemes as well as words clustered into categories (category-based models). The first two approaches are mostly used for agglutinative languages, where words are constituted from sub-word units having their own meaning and syntactic features [3,4]. The last approach might be applied to any language. Its idea is that several words (or morphemes) having similar meaning and syntactic features (e.g. nouns representing cities: *Washington*, *New York*, *Boston*) are substituted by a category, i.e. a words-cluster. As a result the number of distinct n-gram units is reduced and the LM is less sparse.

The advantage of such a language model is that it does not require a large corpus to be generated and since the number of categories might be substantially smaller than the number of words, it allows to use higher-order n-grams.

The method of building the LM presented in this paper follows the category-based approach to language modeling. The grammatical classes of the words are used as their categories. As a result, a sequence of words *Mary killed the bug* is converted to a sequence of a *noun* followed by a *verb*, a *determiner* and another *noun*.

The problem associated with such a language model is that in order to compute the estimated probabilities of n-grams we have to assign the grammatical classes to the words. This is a well known task of part-of-speech (POS) tagging. Although the POS tags might be assigned automatically, in our approach to compute the n-gram probabilities we use a manually tagged training corpus. Although it requires substantial amount of work, for many languages corpora large enough for building the LM are available.

Even though we do not use a POS tagger to build the model, we use this module for tagging the candidate sentences. This tagger is used to tag the n-best candidates generated by the acoustic module in order to compute the POS-sequence probability. In the conducted experiment we use an external component to perform the POS tagging, which heavily incurs the speed of the system. However, it is possible to integrate the POS tagger and optimize it for the ASR task.

The presented experiments were carried out for Polish – an inflectional language with a large number of distinct word-forms. As a result the data sparsity problem of the word-based LMs is more apparent. Recently a large (1 million) corpus of syntactically tagged texts was made available [5] which we used in our experiment. We also used the state-of-the-art POS tagger for Polish [6], which was trained on that corpus. The results for a testing corpus with 100 sentences show that using this LM to rescore the results provided by HTK acoustic model can reduce the word error rate (WER) by more than 10 percentage points.

2 Related Work

Generally, there is very little interest in using POS tags in ASR. Besides [7] and [8], which report negative results on applying POS taggers in ASR, there is little literature on this topic. In [8], a POS tagger [9] was tested as possible improvement in speech recognition of Polish [8]. However, the results were negative because the output of the tagger was too ambiguous.

In the approach described here, we limited the ambiguity by reducing the number of details considered in the model. Specifically we used only grammatical classes of the words, discarding the values of other grammatical categories. A new version of the tagger was applied which could have some impacts as well on the results. However, the difference in efficiency of taggers is probably not big enough to be the main source of the much better efficiency of LM. The tagger used in the experiment from 2008 [8] had an accuracy of 93.44%. The tagger used in the experiment described here [6] has a reported accuracy of 90.34%, which seems to be lower. Still, these numbers are not directly comparable, especially since the reference resources used to evaluate the taggers were different. In the past, it was a frequency dictionary containing approx. 600 thousands of segments, at present it is the rigorously tagged National Corpus of Polish, containing more than 1 million segments [5]. Suffice it to say, that the successor of the tagger used in [8] achieves 87.50% accuracy on that corpus.

3 Resources

The algorithm used to re-score the acoustic hypotheses requires the following resources:

- a corpus with POS tags assigned to words, which is used to generate the LM,
- an inflectional dictionary, which generates possible grammatical categories for the words,
- a POS tagger which selects the most probable grammatical category for each word,
- a test set with the hypotheses provided by the acoustic model used to test the performance of the algorithm.

In the experiment we use a 1-million subcorpus of the National Corpus of Polish (NKJP) [5], which, among other annotations, contains manually selected grammatical classes for all the text segments. The annotations are available in TEI (version 5) standard [10] in XML files. Each individual text is annotated separately on several levels:

- sentence segmentation,
- morphosyntactic tagging,
- semantic tagging,
- shallow syntactic tagging,
- named entities tagging.

The morphosyntactic tagging is the one used to build the LM. The tagging provided in NKJP covers the following phenomena:

- segmentation of the texts into sentences,
- segmentation of the sentences into segments (i.e. units with their own tags attached),
- segment lemmatization,
- grammatical classes of the segments,
- values of grammatical categories of the segments.

In NKJP there are 35 grammatical classes used to classify the segments. These do not correspond directly to the traditional parts-of-speech, such as *nouns*, *verbs* and *adjectives*. Mostly due to the fact that verbs are split into several distinct classes. These classes are described in Table 1.

Although it is easy to obtain the information about the grammatical classes from a single file using XPath queries, there are two technical problems associated with the corpus. The first is its structure – each text (3.9 thousands in total) is annotated in several files and occupies a single directory. As a result the files have to be processed individually, which is not very convenient. This is a bit surprising, since the predecessor of NKJP, IPI PAN corpus [11] was distributed in a binary form, with accompanying corpus server named Poliqarp [12]. The server simplified the access to the data and offered competitive performance.

The second problem is connected with the fact, that although the grammatical classes for each segment are available separately, the decision made by the annotator, i.e. the proper grammatical class of the segment in the given context is only available as a part of string containing the lemma, the class and the values of the grammatical categories concatenated using a colon (:). E.g. for the word *Zatrzasnął* the following tagging is provided: `zatrzasnąć:praet:sg:m1:perf`. In most of the cases this works fine, but there are segments such as *http://www.jeri.gwflota.com/main/*, which become ambiguous when concatenated with the tags. Although, it is possible to isolate the tags from the lemma by subtracting the lemma, which is available separately, it would be much easier to do, if the lemma and the tags were not concatenated.

In order to use the corpus to build the LM, all the grammatical classes found in the corpus were extracted using XPath queries, preserving the sentence segmentation. The sequences of the classes' tags from the sentences were saved in the following lines of a text file. As a result it was possible to use standard LM building tools, such as SRILM [13]. The file consisted of more than 86 thousands of lines and more than 1228 thousands of tokens. The statistic of the unigram counts are given in Table 1 (cf. [5] for the description of the tags).

The second resource used in the experiment is an inflectional dictionary. We use Morfeusz, which contains more than 200 thousands of lexemes and is distributed in the form of a finite state transducer [14]. The dictionary contains very broad general Polish vocabulary, but lacks proper names. From the point of view of speech processing, the most important feature of the dictionary is how it segments the words in a running text.

Although Polish is an inflectional language with several agglutinative features, the dictionary is quite conservative in identifying words as segments, with one important exception: the first and the second person of the past forms of verbs are split into two segments. E.g. the form *jadłem* ((I) ate) is divided into two separate segments: the core *jadł*, indicating the gender and the number of the verb and the agglutinative suffix *em*, indicating the first person. Although this is motivated by the fact, that the suffix might be attached to almost any word in the sentence and that there are several other such suffixes (mostly particles) this complicates the integration of the inflectional dictionary into the ASR framework.

It contrasts with the fact that in most of the cases the agglutinative suffix of the verb is attached to the verb. In the set of sentences used to test the performance of the model containing more than 100 examples, there were only 3 sentences with the agglutinative suffix and all of them were attached to the verb. Taking into account that the inclusion of this phenomenon would considerably complicate the model and its relatively low probability, we decided to exclude these sentences from the test set.

The third resource that was used in the experiment was the POS tagger. We decided to use the state-of-the art tagger, namely WCRFT – a tiered conditional random fields tagger [6]. This tagger is very flexible and might be used with a number of tagsets, however, the preparation of the training data as well as training of the model takes substantial amount of time, we decided to use the

Table 1. Unigram statistic of Polish parts-of-speech

POS	**Grammatical class**	**Tag**	**Count**	**P**
Noun	regular n.	subst	306 236	0.24929
Punctuation		interp	221 699	0.18048
Adjective	regular adj.	adj	125 559	0.10221
Preposition		prep	91 928	0.07483
Conjunction	coordinate conj.	conj	75 513	0.06147
Verb	finite form of v.	fin	60 164	0.04898
Verb	participle-like form of v.	praet	53 759	0.04376
Adverb		adv	51 960	0.04230
Kublik		qub	38 169	0.03107
Unknown word		ign	36 529	0.02974
Interjection		interj	28 372	0.02310
Conjunction	subordinate conj.	comp	22 753	0.01852
Verb	infinitive form of v.	inf	19 606	0.01596
Verb	passive participle	ppas	13 387	0.01090
Verb	gerund	ger	11 842	0.00964
Pronoun	3rd-person personal pron.	ppron3	11 476	0.00934
Pronoun	non-3rd-person personal pron.	ppron12	8 212	0.00669
Abbreviation		brev	8 200	0.00668
Numeral	cardinal	num	8 082	0.00658
Verb	agglutinative form of v.	aglt	7 654	0.00623
Verb	active participle	pact	5 587	0.00455
Verb	predicative form of v.	pred	3 973	0.00323
Verb	future form of the verb „to be”	bedzie	2 804	0.00228
Verb	present participle	pcon	2 644	0.00215
Verb	imperative form of v.	impt	2 524	0.00205
Noun	depreciative n.	depr	2 456	0.00200
Pronoun	reflexive pron.	siebie	2 142	0.00174
Verb	impersonal v.	imps	2 138	0.00174
Verb	„winien”-like verb	winien	813	0.00066
Burkinostka		burk	608	0.00049
Adjective	post-preposition adj.	adjp	580	0.00047
Adjective	pre-adjective adj.	adja	562	0.00046
Verb	perfective participle	pant	154	0.00013
Foreign form		xxx	146	0.00012
Numeral	collective num.	numcol	125	0.00010
Adjective	predicative adj.	adjc	55	0.00004
Total			**1228411**	**1**

model for Polish that is readily available for the tagger. It was trained on the same 1-million subcorpus of NKJP, that we use to build our POS n-gram model and it works well with the Morfeusz inflectional dictionary. It is reported that it achieves a precision of 90.34%, which makes it the best performing POS tagger for Polish[1].

108 recorded sentences or phrases were used for tests. They were spoken by one male, without any specially added noise, but in an office with working computers etc. The content of the corpus is mixed. There are some sentences with political context, like parts of parliament speeches (but not taken from Parliament transcripts). There are some lyrics of Kaczmarski song and speeches of Piłsudski and Balcerowicz.

HTK [17,18] was used to provide n-best list (limited to 600 of items) of acoustic hypotheses for sentences from the test corpus. The acoustic model was trained on CORPORA [19], which means that different speakers, sentences and recording devices were used than for the test set. The hypotheses were constructed as combinations of any words from the corpus as ordered lists of words. This model was trained in a way which allowed all possible combinations of all words in a dictionary to have more variations and to give opportunity for a language model to improve recognition.

4 Rescoring Algorithm

The general idea of the rescoring algorithm is as follows: when the acoustic model generates the n-best list of candidates, each candidate sentence is tagged with the POS tagger. Then the LM-based probability of the sequence of tags is computed and the hypotheses are scored according to the following equation:

$$P(h_i) = P(h_i)_{LM}^{\alpha} * P(h_i)_{AM}^{1-\alpha} , \tag{1}$$

where:

- $P(h_i)$ – the probability of the i-th hypothesis,
- $P(h_i)_{LM}$ – the probability of the i-th hypothesis according to the language model,
- $P(h_i)_{AM}$ – the probability of the i-th hypothesis according to the acoustic model,
- α – the weight of the LM component.

The weighting factor α is introduced, since the LM assigns much higher probabilities to the sequences, because it picks only one class out of 38 (35 grammatical classes + out-of-vocabulary words + start and end of a sentence), while the AM module have thousands of words to consider.

[1] In the past it was reported that several POS taggers of Polish achieve better POS tagging performance. However Radziszewski uses a refined methodology for computing the tagger performance so the results are not comparable as such. In the paper cited the results are compared directly for WMBT [15], PANTERA [16] and WCRFT.

Table 2. Some sentences from the test corpus with tags provided by WCRFT and their English translations

Platforma obywatelska wymaga funkcjonowania klubu w czasie obrad sejmu. subst adj fin ger subst prep subst subst subst interp Civic Platform expects the club to operate during parliament proceedings.
Łatwo skierować czynności do sądu. adv inf subst prep subst interp It is easy to move actions to court.
Wniosek rolniczego związku znajduje się w ministerstwie. subs adj subst fin qub prep subst interp The petition of the agricultural union is in the ministry.
Projekt samorządu ma wysokie oczekiwania finansowe. subst subst fin adj subst adj interp The municipality project has high financial expectations.
Fundusz społeczny podjął działania w ramach obecnego prawa cywilnego. subst adj praet subst prep subst adj subst adj interp The communal foundation took steps according to existing civil law.
Uchwała rządowa dotycząca handlu i inwestycji przedsiębiorstw państwowych w rynek nieruchomości. subst adj pact subst conj subst subst adj prep subst subst interp The government act on trade and investments of public enterprises in the estate market.
Panie marszałku, wysoka izbo. subst subst interp adj subst interp Mr speaker, House. (common way to start a speech in the Polish Parliament)
Bezpieczeństwo jest bardzo ważne. subst fin adv adj interp The safety is very important.
Skrzydła im ścierpły w długiej niewoli. subst ppron3 praet prep adj subst interp Their wings went numb in a long captivity.
Chcą być przeklęci pierwsi. fin subst adj adj interp They want to be cursed first.
Świat odkrywa na nowo wciąż dramaty moje. subst fin prep adv adj subst adj interp The world discovers my drama all over from the beginning.
Spały wilczki dwa zupełnie ślepe jeszcze. praet subst num adv adj qub interp Two baby wolfs slept completely blind.
Zajmują ją w imieniu władzy naczelnej rządu narodowego. fin ppron3 prep subst subst adj subst adj interp They take it with authority of the supreme authority of the national government.
Socjalizm zostawił w Polsce w owym roku spodlony pieniądz. subst praet prep subst prep adj subst ppas subst interp Socialism left in Poland that year degraded money.

The probability assigned by the LM is computed using the part-of-speech n-grams collected from the NKJP subcorpus. In general the probability of a given sequence of tags using n-grams is computed as the maximum likelihood estimation (MLE):

$$P(v_i|v_{i-N+1}...v_{i-1}) = \frac{c(v_{i-N+1}...v_i)}{c(v_{i-N+1}...v_{i-1})} , \tag{2}$$

where:

- $P(v_i|v_{i-N+1}...v_{i-1})$ – is the probability of the category v_i assuming the sequence of $v_{i-N+1}...v_{i-1}$ previous categories,
- $c(v_{i-N+1}...v_i)$ – is the number of sequences observed in the corpus, consisting of categories from v_{i-N+1} to v_i.

However, the direct application of MLE faces the problem of sequences that have never been seen in the training data. Their count $c(v_{i-N+1}...v_i)$ equals zero and their probability is also 0. As a result, the whole sequence has 0 probability. There are many methods used to overcome this problem, namely [20,21]:

- smoothing,
- interpolation,
- backoff.

Smoothing assigns the probability of unseen n-grams directly by estimating it using the n-grams with low frequency (e.g. n-grams which occurred only once). To compensate the mass of probability that was distributed to the unseen n-grams it discounts the counts of the n-grams that have been seen in the corpora. In interpolation lower order n-grams are combined linearly and the probability is non-zero, at least if the token in question have ever been seen in the training data. The last method – backoff – uses strategy similar to interpolation, but it uses lower order n-gram only if the count for the given sequence of the length n is 0.

Although Kneser-Ney discounting [22] is the most popular and the best performing method used for large word dictionaries, it does not work if the dictionary is very small, like in the POS-based n-grams[2]. That is why, we use Witten-Bell discounting, namely the backoff version of this method.

To define the probability of a sequence of tags, we first define the discounted frequency

$$F(v_{j-N+1}...v_j) = \frac{c(v_{j-N+1}...v_j)}{n(v_{j-N+1}...*) + c(v_{j-N+1}...v_{j-1})} , \tag{3}$$

where $n(v_{j-N+1}...*)$ – the number of sequences of length n with the prefix $v_{j-N+1}...v_{j-1}$ that appeared only once in the corpus.

Then the probability of a sequence of tags is estimated as

$$P(v_j|v_{j-N+1}...v_{j-1}) = \begin{cases} F(v_{j-N+1}...v_j) & c(v_{j-N+1}...v_j) > 0 \\ \beta P(v_j|v_{j-N+2}...v_{j-1}) & otherwise \end{cases} , \tag{4}$$

[2] Cf. http://www.speech.sri.com/projects/srilm/manpages/srilm-faq.7.html

where $\beta = \frac{1-\sum F(v_{j-N+1}...v_j)}{1-\sum F(v_{j-N+2}...v_j)}$ – the backoff weight.

The final probability of a sentence computed using the LM is

$$P(h_i)_{LM} = \prod_{w_j \in h_i} P(V(w_j)|V(w_{j-N+1})...V(w_{j-1})) \ , \tag{5}$$

where $V(w_j)$ is the grammatical class assigned to the word w_j.

It is assumed that the grammatical categories corresponding to the words present in the sentence are fully determined. This assumption is accomplished by incorporating the POS tagger into the system. Although many of the words have several possible interpretations defined in the inflectional dictionary, only one of them is selected according to the tagger. The remaining options are not taken into account.

The ranking of the hypotheses is defined as follows

$$\begin{gathered} R(h_i) = logP(h_i) = \alpha log(P(h_i)_{LM}) + (1-\alpha)log(P(h_i)_{AM}) = \\ \alpha \sum_{w_j \in h_i} P(V(w_j)|V(w_{j-N+1})...V(w_{j-1})) + (1-\alpha) \sum_{w_j \in h_i} P(w_j|s)_{AM} \ , \end{gathered} \tag{6}$$

where $P(w_j|s)_{AM}$ is the probability of the word w_j conditioned on the speech signal s, computed by the acoustic module. The ranking is computed in log-space, since the acoustic probabilities are very low and are subject to underflow errors.

The value of the parameter α might be optimized on the held out corpus.

5 Results

To measure the performance of the POS-based LM we used the following measures:

- word error rate reduction (WERR),
- correct sentence position improvement (CSPI) in the n-best list of hypotheses.

Word error rate is defined as the minimum edit distance [21, p. 73-77] between the correct sentence and the hypothesis with the highest probability. In our setting each edition has the same cost. Word error rate reduction is the number of percentage points the word error rate has reduced after applying the LM.

Table 3. The performance of the POS-based language model

N	**WERR**$_{best}$ %	**CSPI**$_{best}$	**WERR**$_{1/2}$%	**CSPI**$_{1/2}$
1	12.55	20.28	2.42	1.25
3	12.61	38.61	5.12	30.93
5	12.69	40.03	5.14	31.57

Table 4. The most popular 3-grams of POS tags in Polish

POS tag 3-gram	%	POS tag 3-gram	%
`subst interp </s>`	3.02	`subst interp subst`	0.69
`adj subst interp`	1.58	`adj subst subst`	0.64
`subst subst interp`	1.26	`subst subst subst`	0.62
`subst prep subst`	1.13	`subst conj subst`	0.53
`prep subst interp`	1.12	`interp interp </s>`	0.53
`prep adj subst`	0.94	`prep subst adj`	0.51
`subst adj interp`	0.89	`subst subst adj`	0.50
`prep subst subst`	0.77	`interp subst interp`	0.49
`subst adj subst`	0.72	`subst interp conj`	0.46
`adj interp </s>`	0.69	`subst interp adj`	0.45

CSPI is defined as the improvement in position of the correct sentence between the n-best list generated by the acoustic model and the list generated by the combined language and acoustic models.

The results of the experiments carried out on the testing corpus are presented in Table 3. They are reported for cases where AM and LM had the same weight (after scaling the probability in order to compensate the difference in the size of AM and LM): $WERR_{1/2}$, $CSPI_{1/2}$ and for the best cases (i.e. for the optimal value of the parameter α): $WERR_{best}$, $CSPI_{best}$. The average WER achieved by the sole AM was 37.23% and the average position of the correct hypothesis was 50 (among 600 hypotheses).

The best WERR achieved for 1-grams, 3-grams and 5-grams is very similar, but it should be noted, that it is obtained for the specifically selected parameter α. The CSPI measure shows large differences in the performance between 1-grams and 3-grams. It is especially apparent when the weight for the LM is the same as for the AM. The difference between the 3-gram LM and 5-gram LM is much smaller – both in the case of the LM with the best parameter and with the parameter set to a predefined value.

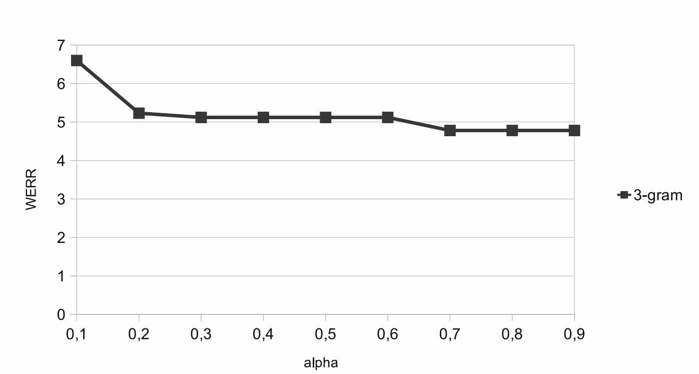

Fig. 1. Word error rate reduction (WERR) for different values of the parameter α in range 0-1.0

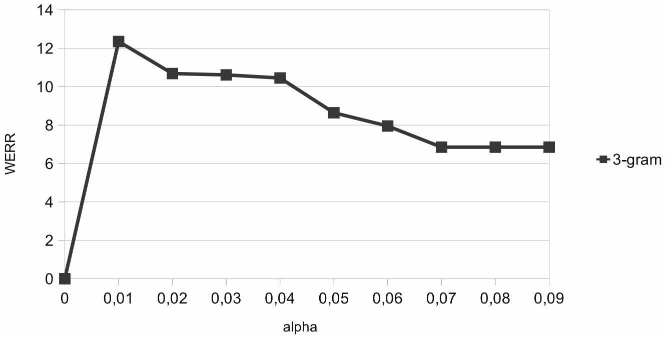

Fig. 2. Word error rate reduction (WERR) for different values of the parameter α in range 0-0.1

Figures 1 and 2 show the dependence of the performance of the rescoring algorithm on the parameter α. The proper selection of the parameter is crucial for the performance of the algorithm.

In Table 4 we also report the POS trigrams that are the most popular in Polish.

6 Conclusions

We conclude that the simplified POS tags are very good source of information for statistical language models of Polish. Applied on 108 test sentences recognized acoustically by HTK with a fixed weight they reduced WERR by 5% points and with the optimal weight – over 12 % points.

Assuming that a manually tagged corpus is available, a POS-based LM is much easier and cheaper to build than a word-base LM. This is particularly important for the inflected languages like Polish.

In the following research we are planning on integrating the POS-based model into a larger ASR system.

Acknowledgments. This work was supported by LIDER/37/69/L-3/11/ NCBR/2012 grant.

References

1. Ziółko, B., Skurzok, D.: N-grams model for Polish. Speech and Language Technologies, Book 2, pp. 107–127. InTech Publisher (2011)
2. Jurafsky, D., Martin, J.H.: Speech and Language Processing, 2nd edn. Prentice-Hall, Inc., New Jersey (2008)
3. Hirsimaki, T., Pylkkonen, J., Kurimo, M.: Importance of high-order n-gram models in morph-based speech recognition. IEEE Transactions on Audio, Speech and Language Processing 17(4), 724–732 (2009)
4. Sak, H., Saraçlar, M., Gungor, T.: Morpholexical and discriminative language models for turkish automatic speech recognition. IEEE Transactions on Audio, Speech, and Language Processing 20(8), 2341–2351 (2012)

5. Szałkiewicz, Ł., Przepiórkowski, A.: Anotacja morfoskładniowa. In: Narodowy Korpus Języka Polskiego, pp. 59–96. Wydawnictwo Naukowe PWN (2012)
6. Radziszewski, A.: A tiered CRF tagger for polish. In: Bembenik, R., Skonieczny, Ł., Rybiński, H., Kryszkiewicz, M., Niezgódka, M. (eds.) Intelligent Tools for Building a Scientific Information Platform. SCI, vol. 467, pp. 215–230. Springer, Heidelberg (2013)
7. Niesler, T., Whittaker, E., Woodland, P.: Comparison of part-of-speech and automatically derived category-based language models for speech recognition. In: Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 1, pp. 177–180. IEEE (1998)
8. Ziółko, B., Manandhar, S., Wilson, R.C., Ziółko, M.: Language model based on pos tagger. In: Proceedings of SIGMAP 2008 the International Conference on Signal Processing and Multimedia Applications, Porto (2008)
9. Piasecki, M.: Hand-written and automatically extracted rules for polish tagger. In: Sojka, P., Kopeček, I., Pala, K. (eds.) TSD 2006. LNCS (LNAI), vol. 4188, pp. 205–212. Springer, Heidelberg (2006)
10. Burnard, L., Sperberg-McQueen, C.: Guidelines for electronic text encoding and interchange. In: Association for Computers and the Humanities, Association for Computational Linguistics, Association for Literary and Linguistic Computing (1994)
11. Przepiórkowski, A.: Korpus IPI PAN. Wersja wstępna. Instytut Podstaw Informatyki PAN (2004)
12. Janus, D., Przepiórkowski, A.: Poliqarp 1.0: Some technical aspects of a linguistic search engine for large corpora. In: The Proceedings of Practical Applications of Linguistic Corpora (2005)
13. Stolcke, A., et al.: SRILM-an extensible language modeling toolkit. In: Proceedings of the International Conference on Spoken Language Processing, vol. 2, pp. 901–904 (2002)
14. Saloni, Z., Woliński, M., Wołosz, R., Gruszczyński, W., Skowrońska, D.: Słownik gramatyczny języka polskiego (Eng. Grammatical dictionary of Polish) (2102)
15. Radziszewski, A., Śniatowski, T.: A memory-based tagger for polish. In: Proceedings of the 5th Language & Technology Conference, Poznań (2011)
16. Acedański, S.: A morphosyntactic brill tagger for inflectional languages. In: Loftsson, H., Rögnvaldsson, E., Helgadóttir, S. (eds.) IceTAL 2010. LNCS, vol. 6233, pp. 3–14. Springer, Heidelberg (2010)
17. Young, S.: Large vocabulary continuous speech recognition: a review. IEEE Signal Processing Magazine 13(5), 45–57 (1996)
18. Young, S., Evermann, G., Gales, M., Hain, T., Kershaw, D., Moore, G., Odell, J., Ollason, D., Povey, D., Valtchev, V., Woodland, P.: HTK Book. Cambridge University Engineering Department, UK (2005)
19. Grocholewski, S.: CORPORA - speech database for Polish diphones. In: Proceedings of Eurospeech (1997)
20. Chen, S.F., Goodman, J.: An empirical study of smoothing techniques for language modeling. In: Proceedings of the 34th Annual Meeting on Association for Computational Linguistics, pp. 310–318. Association for Computational Linguistics (1996)
21. Jurafsky, D., Martin, J., Kehler, A.: Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition, 2nd edn. Prentice Hall (2009)
22. Kneser, R., Ney, H.: Improved backing-off for m-gram language modeling. In: 1995 International Conference on Acoustics, Speech, and Signal Processing, ICASSP 1995, vol. 1, pp. 181–184. IEEE (1995)