SUB-WORD MODELING OF OUT OF VOCABULARY WORDS IN SPOKEN TERM DETECTION

Igor Szöke, Lukáš Burget, Jan Černocký, Michal Fapšo

Speech@FIT, Faculty of information Technology, Brno University of Technology, Czech Republic
szoke@fit.vutbr.cz

ABSTRACT

This paper deals with comparison of sub-word based methods for spoken term detection (STD) task and phone recognition. The sub-word units are needed for search for out-of-vocabulary words. We compared words, phones and multigrams. The maximal length and pruning of multigrams were investigated first. Then two constrained methods of multigram training were proposed. We evaluated on the NIST STD06 dev-set CTS data. The conclusion is that the proposed method improves the phone accuracy more than 9% relative and STD accuracy more than 7% relative.

Index Terms— phone, multigram, spoken term detection, sub-word, keyword spotting, syllable, lattice

1. INTRODUCTION

Spoken term detection (STD) is an important part of speech processing. Its goal is to detect terms in spoken documents, such as broadcast news, telephone conversations, or meetings. The most common way to perform STD is to use the output of large vocabulary continuous speech recognizer (LVCSR). Rather than using the 1-best output of LVCSR, the state-of-the-art STD systems search terms in lattices - acyclic oriented graphs of parallel hypothesis. In addition to better chances to find the searched term, the lattices also offer to estimate the confidence of given query [2].

A drawback of the LVCSR system is, that it recognizes only words which are in an LVCSR vocabulary, so that the following STD system can not detect out-of-vocabulary words (OOVs). On the other hand, OOVs usually carry lot of information (named entities, etc.). Common way to search OOVs is to use subword units – a search term is converted into a sequence of subword units when it is entered, either using a dictionary (which can be much larger than that of LVCSR) or by a grapheme to phoneme (G2P) converter based on linguistically based or automatically trained rules. Such sequence is then searched in the output of a subword recognizer.

The most intuitive subword units are phones. Another type of subword units are syllables, phone n-grams, multigrams or broad phonetic classes [1] which are all based on phones. In [1], phone n-grams with fixed length n were used for indexing of phone strings or lattices. The optimal length of phone n-grams was found to be 3.

In our prior work [6], we have used sequences of overlapped 3-grams for search. However, words shorter than 3 phones must be processed in a different way or dropped. Another drawback of the fixed length is that the frequencies of units are not taken into account although some units are more frequent than others. **Variable length units** can be used to overcome this problem: a rare unit is split into more frequent shorter units while a frequent unit can be represented as a whole. The other advantage is that variable length units can reflect word sequences and compensate for missing word language model.

Disk space and computational requirements are also important from practical point of view – stored data and search time must be kept as small as possible. The decoding should be fast and an indexing must be used. The tradeoff between index size and search accuracy must be also included to the evaluation.

In this paper, we investigate into the use of *multigrams* as subword units. We study, which impact multigram parameters have on the accuracy and index size. We try to find the optimal length and pruning of multigram units. We also propose two improvements of multigram training algorithm to reflect word boundaries. Another benefit is the incorporation of standard n-gram language model on the top of multigram units. We compare proposed multigrams to standard phone lattices and word-based STD system, and we evaluate phone accuracy, STD accuracy and the size of index.

2. MULTIGRAMS

The multigram language model was proposed by Deligne et al. [3]. Multigram model is a statistical model having sequences with variable number of units. The definition of multigram model and its parameter estimation follows:

Let $U = u(1) \dots u(T)$ denote a string of T units, and let S denote a possible segmentation of U into q sequences $q \leq T$ of units $s(1) \dots s(q)$. The n-multigram model computes the joint likelihood $L(U, S)$ of the corpus U associated to segmentation S as the product of the probabilities p of the successive sequences, each of them having a maximum length of n:

$$L(U, S) = \prod_{t=1}^{t=q} p(s(t)) \tag{1}$$

Denoting as $\{S\}$ the set of all possible segmentations of U into sequences of units, the likelihood of U is:

$$L_{mgr}^{best}(U) = \max_{S \in \{S\}} L(U, S) \tag{2}$$

A n-multigram model is fully defined by a set of parameters P consisting of the probability of each unit sequence $S_i \in D$ in a dictionary $D = \{s_1; \dots s_m\}$ that contains all the sequences which can be formed by combination of $1, 2, \dots, n$ units:

$$P = (p_i)_{i=1}^{m} \quad \text{where} \quad p_i = p(s_i) \quad \text{and} \quad \sum_{i=1}^{m} p_i = 1 \tag{3}$$

ML estimates of P can be computed through an Viterbi algorithm iteratively. Let $S^{*(k)}$ denote the most likely segmentation of U with given parameters P^k at iteration k:

$$S^{*(k)} = \arg\max_{S \in S} L(S|U, P^k) \tag{4}$$

According to [3], the re-estimation formula of i^{th} parameter (sequence) at iteration $k+1$ is defined as

$$P_i^{k+1} = \frac{c(s_i, S^{*(k)})}{c(S^{*(k)})}, \quad (5)$$

where $c(s_i, S)$ is the number of occurrences of sequence s_i in segmentation S and $c(S)$ is the total number of sequences in S. The set of parameters P is initialized with the relative frequencies of all occurrences of units up to length n in the training corpus. To avoid overlearning, it is advantageous to discard low probable sequences: by setting $p_i = 0$ to all $c(s_i) \leq c_0$. The c_0 parameter is denoted as multigram pruning parameter. Sequences of length $n = 1$ are excluded from pruning to ensure that each sequence is segmentable. If a unit with length $n = 1$ has 0 occurrences in S, then it's probability is set to a very low number.

3. SYSTEM DESCRIPTION

During the pre-processing, the acoustic data was split into shorter segments in silences (output of speech/nonspeech detector) longer than 0.5s. The data was also split if the speaker changed (based on the output of diarization). Segments longer than 1 minute were split into 2 parts in silence the closest to the center of the segment. This was done to overcome long segments and accompanying problems during decoding.

Acoustic models from an LVCSR system were used for subword recognition. Presented LVCSR system was derived from AMIDA project LVCSR [7]. Our LVCSR operates in 3 passes:

In the first pass, the front-end converts the segmented recordings into feature streams, with vectors comprised of 12 MF-PLP features and raw log energy, first and second order derivatives are added. After, a cepstral mean and variance normalization (CMN/CVN) is performed on a per channel basis with given segmentation. The first decoding pass yields initial transcripts that are subsequently used for estimation of VTLN warp factors. The feature vectors and CMN and CVN are recomputed. The models were basic ML trained HMMs.

The second pass processes the new features and its output is used to adapt models with maximum likelihood linear regression (MLLR). Bigram lattices are produced and re-scored by trigram and fourgram language model. Acoustic models were VTLN HLDA MPE trained HMM.

In the third pass, posterior features [5] are generated. The output from second pass is used to adapt models with Constrained MLLR (CMLLR) and MLLR. The bigram lattices with posterior features are produced. Acoustic models were VTLN LC-RC SAT MPE trained.

All systems use standard cross-word tied states HMM.

The acoustic models are trained on *ctstrain04* corpora which is a subset of *h5train03* set defined at Cambridge. Total amount of data is 277 hours. A bigram word language model (used for comparison) was trained on 1492.5M words of a mix of 12 corpora. See [5] for details.

The same *ctstrain04* corpora was used as base phone corpora for our experiments. The size is 11.5M phones. In subword recognition, only the recognition network (language model and unit dictionary) changed.

The term confidence measure, which is produced by the term detector, is the posterior probability of the term. If the term has one word, the confidence is the posterior probability of the word (link) in a lattice.

3.1. Subword training data

The phones and multigrams are trained on phone strings. N-gram language model is trained for phone system. Multigram training has 2 steps. Multigram dictionary and unit probabilities are estimated in the first step. Standard n-gram language model is then estimated in the second step.

The *ctstrain04* was searched for utterances containing "out-of-vocabulary" words defined in section 4. These utterances were omitted (denoted *LnoOOV*). Also, a corpus containing OOVs (denoted as *LOOV*) and having the size same as *LnoOOV* was derived from *ctstrain04*.

According to size of *LOOV* and the iterative multigram training procedure, the data used for estimation of multigram dictionary was reduced to 3.75M phones to achieve reasonable training time (several hours). This corpus was denoted as *MOOV*. Corresponding corpus derived from *LnoOOV* was also created and denoted as *MnoOOV*. The sizes of above mentioned corpora are summarized in table 1.

Notation	# of utters.	# of phones (incl. sil)	# of phones (w/o sil)
LnoOOV	237.2K	6.40M	5.60M
LOOV	169.1K	6.26M	5.60M
MnoOOV	143.5K	3.82M	3.35M
MOOV	106.4K	3.75M	3.35M

Table 1. Comparison of corpora used for multigram dictionary (M*) and language model (L*) training.

4. EVALUATION

Because NIST has not released the STD06 eval data, we evaluate on the STD06 development set (data and term list). The original NIST STD06 development term set for CTS contains low number of OOVs. First of all, 124 terms containing true OOVs were omitted. Then, we selected 440 words from the term set and other 440 words from the LVCSR vocabulary. A limited LVCSR system was created (denoted by *WRDRED* which means "reduced vocabulary") where these 880 words were omitted from the vocabulary. This system had reasonably high OOV rate on the NIST STD06 DevSet. The term set has 975 terms of which are 481 in vocabulary (IV) terms and 494 OOV terms (terms containing at least one OOV) for the reduced system. The number of occurrences of the IV terms is 4737 and 196 of OOV terms.

Note, that because of the lack of separate development and test sets, also several parameters are tuned on the development set. It is word/multigram/phone insertion penalty and acoustic scaling factor.

4.1. UBTWV - Upper Bound TWV

We used Term Weighted Value (TWV) for evaluation of spoken term detection (STD) accuracy of our experiments. The TWV was defined by NIST for STD2006 evaluation [4]. One drawback of TWV metric is its one global threshold for all terms. This is good for evaluation for end-user environment, but leads to uncertainty in comparison of different experimental setups. We do not know if the difference is caused by different systems or different normalization and global threshold estimation. This is a reason for *Upper Bound TWV* (UBTWV) definition which differs from TWV in individual threshold for each term. Ideal threshold for each term is found to maximize the term's TWV:

$$thr_{ideal}(term) = \arg\max_{thr} TWV(term, thr) \quad (6)$$

Corpus	LM	Phone		
	n-gram	ACC	UBTWV-ALL	PhnSIZE
LOOV	1	53.84	35.0	10.67M
LOOV	2	58.41	47.4	8.24M
LOOV	3	59.75	48.1	6.45M
LnoOOV	1	53.82	35.0	10.73M
LnoOOV	2	58.41	47.0	8.20M
LnoOOV	3	59.66	48.3	6.38M

Table 2. Comparison of training corpora and n-gram order on accuracy for phone based experiments.

The UBTWV is then defined as

$$UBTWV = 1 - \underset{term}{average}\{p_{MISS}(term, thr_{ideal}(term)) + \beta p_{FA}(term, thr_{ideal}(term))\}, \quad (7)$$

where β is 999.9. It is equivalent to a shift of each term to have the maximal $TWV(term)$ at threshold 0. Two systems can be compared by UBTWV without any influence of normalization and ideal threshold level estimation on the systems TWV score. The *UBTWV* was evaluated for the whole set of terms (denoted *UBTWV-ALL*), only for in-vocabulary subset (denoted *UBTWV-IV* and only for out-of-vocabulary subset (denoted *UBTWV-OOV*.

4.2. Lattice vs. Index Size

Using STD in large scale implies using an indexing technique where the size of index is important. That is why we calculate the lattice size in the same way the index is build: Groups of the same overlapped words are found in the word or multigram lattice. Each group is substituted by one candidate and the count of such candidates is denoted *WrdSIZE*. Phone lattices are not processed phone-by-phone, but by indexing phone trigrams. Phone trigrams are generated first of all. Then the same procedure is applied as for the word lattices: groups of the same phone trigrams are identified and each group is substituted by one candidate. The count of such candidates is denoted *PhnSIZE*.

5. EXPERIMENTS

The first experiment was done with phones. Word insertion penalty and LM scaling factor were tuned giving the best results on the test set. The decoder pruning parameter was set to a value where phone accuracy saturated. UBTWV-ALL can be improved by another 2%, but the PhnSIZE rises exponentially by further softening the pruning. Phone system results are summarized in table 2. We conclude that absence of OOV in the training corpora has no influence on the accuracy. The trigram LM provides the best accuracy but the recognition network is 7 times bigger compared to the bigram network.

The second experiment was done with multigrams. Word insertion penalty, LM scaling factor and the decoder pruning were tuned with respect to the best phone accuracy. UBTWV-ALL can be improved by another 1% absolutely in case of big increase of PhnSIZE/WrdSIZE by further softening the decoder pruning.

The word or multigram recognizer produces word or multigram lattices where terms appear as sequences of words or multigrams. However, the recognizer can be switched to produce phone lattices even if it is word/multigram recognizer. We can evaluate phone accuracy and detection of search terms as phone strings even if such phone lattice is produced by a LVCSR system. We defined *WrdUBTWV* and *PhnUBTWV* to distinguish between word and phone level output of recognizer. The *WrdUBTWV* means that term is a sequence of words/multigrams and is searched in word/multigram lattice. The *PhnUBTWV* means that term was converted to phone sequence and is searched in phone lattice generated by word/multigram system.

Different multigram pruning c_0 was tried in range between 2 and 200. Word insertion penalty and LM scale factor were tuned for each chosen c_0 value. The phone accuracy was not influenced much by the multigram pruning. It saturates for $c_0 = 20$ and both WrdUBTWV and PhnUBTWV saturate around $c_0 = 50$. That is why the multigram pruning factor was chosen 50 in the following experiments.

Next experiment tests accuracies depending on the maximum length of multigram units. We found that phone accuracy and both UBTWVs saturate for multigram units of length 5.

The last experiment deals with the order of the used n-gram language model (table 5). The conclusion is that trigram is the best. The bigram is not much worse but the size of bigram recognition network is $1/2$ of the trigram network, so the decoding is faster.

6. CONSTRAINED MULTIGRAM UNITS

The baseline process of building multigram unit dictionary is without any constraints (denoted *xwrd*). The corpus of phone strings is taken as is. An example of an utterance segmented by such unconstrained units is in table 3 line 2. A multigram unit can be placed across word boundaries and also across silences (`sil`). Incorporation of word boundaries (cross word multigrams) into multigram units means, that multigrams also somehow reflect the word language model. The question is whether this is good or not. The same question can be asked about the `sil`. By incorporating silence into multigrams, the units are learned to remember parts of speech where silence is usual and where it is not. Two experiments with constrained training of multigram dictionary were done to evaluate the influence of cross-word multigrams and silence inside a multigram unit:

6.1. No Silence in Multigram

Only multigram units which do not contain silence were trained in this experiment (denoted *nosil*). The unigram `sil` unit is the only one unit which contains silence. This is needed to make utterances segmentable. An example of an utterance segmented by this *nosil* method is in table 3 line 3.

6.2. Non Cross-word Multigrams

Word boundaries were marked in the training corpus. Then the following rule was incorporated into the training algorithm: word boundary will appear at most at the beginning or at the end of a multigram unit. Only two units (ending and starting) with the word boundary marker can be put besides each other during the segmentation. This system is denoted as *noxwrd*. An example of a utterance segmented by *noxwrd* method is in table 3 line 4. The word boundary marker is denoted by a star.

The multigram parameters for the following experiments were: maximum multigram length 5, multigram pruning 50, multigram training corpus *MnoOOV* and language model training corpus *LnoOOV*. The decoder parameters were the same as for the previous multigram experiments.

The results comparing multigrams trained with different constraints are summarized in table 5. Three orders of n-gram language model were trained for each of these three systems. The *"Phone"* column contains results when the multigram decoder was switched to produce phone lattices.

word	sil YEAH I MEAN IT IS sil INTERESTING									
xwrd	sil-y-eh-ax		ay-m-iy-n		ih-t-ih-z-sil			ih-n-t-ax-r	eh-s-t-ih-ng	
nosil	sil	y-eh-ax	ay-m-iy-n		ih-t-ih-z		sil	ih-n-t-ax-r	eh-s-t-ih-ng	
noxwrd	*sil*	*y-eh-ax*	*ay*	*m-iy-n*	*ih-t*	*ih-z*	*sil*	*ih-n	t-ax-r-eh-s	t-ih-ng*

Table 3. Examples of different multigram segmentations.

Unit	LM	Phone	UBTWV			SIZE
	n-gram	ACC	ALL	IV	OOV	
WRDREDwrd	2	-	51.4	73.4	0.00	0.56Mw
WRDREDphn	2	65.40	54.0	55.4	50.8	4.34Mp
phn-LnoOOV	3	59.66	48.3	45.3	55.2	6.38Mp
mgram-xwrd	3	65.25	55.9	55.2	57.7	1.4Mw/3.6Mp
mgram-nosil	3	65.42	58.4	57.8	59.7	1.2Mw/4.1Mp
mgram-noxwrd	3	65.10	**63.0**	64.7	59.3	1.7Mw/3.7Mp

Table 4. Comparison of word, phone and multigram systems.

System	LM	Phone			Multigram	
	ngram	AC	UBTWV-ALL	SIZE	UBTWV-ALL	SIZE
xwrd	1	60.33	51.9	20.0M	53.7	3.5M
xwrd	2	63.13	53.3	9.4M	**56.8**	2.0M
xwrd	3	**65.25**	**54.7**	3.6M	55.9	1.4M
nosil	1	60.38	53.1	21.6M	54.6	3.2M
nosil	2	63.25	**55.1**	9.3M	**59.2**	1.8M
nosil	3	**65.42**	54.7	4.1M	58.4	1.2M
noxwrd	1	59.30	50.1	20.3M	53.5	7.3M
noxwrd	2	62.87	52.8	7.6M	61.3	3.1M
noxwrd	3	**65.10**	**54.1**	3.7M	**63.0**	1.7M

Table 5. Comparison of accuracy of xwrd, nosil and noxwrd multigram systems. All systems have length 5, pruning 50 and were trained on *MnoOOV* and *LnoOOV*.

System	trained	Phone		Multigram
		AC	UBTWV-ALL	UBTWV-ALL
xwrd	OOV	64.32	56.1	56.6
nosil	OOV	65.84	57.6	60.3
noxwrd	OOV	65.50	55.6	67.4
xwrd	noOOV	65.25	54.7	55.9
nosil	noOOV	65.42	54.7	58.4
noxwrd	noOOV	65.10	54.1	63.0

Table 6. Comparison of accuracy of xwrd, nosil and noxwrd multigram systems trained on corpora including (*MOOV*, *LOOV*) and excluding (*MnoOOV*, *LnoOOV*) the OOVs. All systems have length 5, pruning 50 and trigram language model.

We conclude that multigram performs better on the "word" level than on the phone level in the STD task. The best accuracy was achieved by *noxwrd* trained multigrams with trigram language model. The best phone accuracy was achieved by *nosil* trained multigrams and bigram language model. The constraints during the training have no significant effect on phone accuracy. On the other hand, the *nosil* units perform slightly better. The impact is on the STD task, where the improvement is about 6% absolute (10% relative).

Terms must be converted from words to multigram sequences for multigram based STD. We also evaluated the impact of term segmentation to the UBTWV. We tried to search 1-best up to all possible segmentations of the terms. Our conclusion is that this has small effect on the accuracy. The accuracy improvement was about several tenth of percent and saturates for the 3-best segmentations of term.

Comparison of multigram systems trained with and without OOVs is in table 6. The OOVs cause a hit of 0.8% relative on phone accuracy and up to 8% relative on the WrdUBTWV-ALL.

7. CONCLUSION

Table 4 compares word, phone and multigram based systems from phone and spoken term detection accuracy point of view. The *WRDREDwrd* was the LVCSR (with reduced vocabulary) on the word level (terms are word sequences). The *WRDREDphn* was the LVCSR switched to phone level. The best phone accuracy was achieved by the *nosil* constrained multigrams. However, better STD accuracy was achieved by the *noxwrd* constrained multigrams. It is important to mention that multigram lattices are significantly smaller and the recognition network is approximately the same size compared to phones.

8. ACKNOWLEDGEMENTS

This work was partly supported by European project AMIDA (FP6-033812), by Grant Agency of Czech Republic under project No. 102/08/0707 and by Czech Ministry of Education under project No. MSM0021630528. The hardware used in this work was partially provided by CESNET under project No. 201/2006.

9. REFERENCES

[1] Ng, K., Zue, V., "Subword-based approaches for spoken document retrieval", Speech Communication, vol 32., no. 3, Oct. 2000, pp. 157-186.

[2] Jiang, H.,"Confidence Measures for Speech Recognition: A Survey", in Speech Communication, 455–470, 2005, vol. 45.

[3] Deligne, S. and Bimbot F." Language modeling by variable length sequences: theoretical formulation and evaluation of multigrams", in proc. of ICASSP, 169–172, May 1995, vol. 1.

[4] Fiscus, J., Ajot, J., and Doddington, G., "The Spoken Term Detection (STD) 2006 Evaluation Plan", NIST USA, Sep 2006.

[5] Szöke, I. et al., BUT System for NIST STD 2006 - English, available from http://www.fit.vutbr.cz/speech/std/2006/, file but_06_std_eval06_eng_all_spch_p-BUT-STBU-MERGED_1.txt

[6] Černocký, J. et al., "Search in speech for public security and defense", in proc. of IEEE Workshop on Signal Processing Applications for Public Security and Forensics, 2007 (SAFE '07), 1–7.

[7] Hain, T. et al, "The AMI Meeting Transcription System", in proc. of NIST Rich Transcription 2006 Spring Meeting Recognition Evaluation Worskhop, Washington D.C., USA, 2006, p. 12.