RECOGNITION OF PHONEMES AND WORDS IN SINGING

Annamaria Mesaros, Tuomas Virtanen *

Tampere University of Technology
Department of Signal Processing
Korkeakoulunkatu 1, 33720, Tampere, FINLAND

ABSTRACT

This paper studies the influence of n-gram language models in the recognition of sung phonemes and words. We train uni-, bi-, and trigram language models for phonemes and bi- and trigrams for words. The word-level language model is estimated from a textual lyrics database. In the recognition we use a hidden Markov model based phonetic recognizer adapted to singing voice. The models were tested on monophonic singing and on vocal lines separated from polyphonic music. On clean singing the phoneme recognition accuracies varied from 20% (no language model) to 39% (bigram) and on polyphonic music from 6% (no language model) to 20% (bigram). In word recognition, one fifth of the words were recognized in clean singing, the performance being lower on polyphonic music. We study the use of the recognition results in a query-by-singing application. Using the recognized words, we retrieve the songs by searching for the text in a text lyrics database. For the word recognition system having only 24% correct recognition rate, the first retrieved song is correct in 57% of the test cases.

Index Terms— singing recognition, speech recognition, query-by-singing

1. INTRODUCTION

Music contains many kinds of information that can be used for automatically analyzing its content. The information given by the lyrics is an important cue in retrieving or classifying songs. For a long time, the use of lyrics for finding songs was limited to performing text search in lyrics databases. When the first content-based retrieval systems appeared, the query by humming/singing applications used only the melody information [1]. Recognizing the semantic content of a sung query can speed up the search by offering a number of hypotheses to narrow down the number of songs for melody search. The transcription (recognition) of lyrics using a large vocabulary speech recognizer (LVCSR) is still regarded as a nearly impossible task because of many aspects. First of all, the performance of automatic speech recognition using a LVCSR is limited. Second, there are important phonetic and timing differences between speech and the singing voice, that must be dealt with. Last but not least, real world music is polyphonic. Even having a system that can recognize singing, the interference of the instrumental background would degrade significantly its performance.

To make the recognition task easier, more or less restrictive language models can be used to control the output of the system by imposing certain sequences of words. Such work was done by Suzuki et al [2] to retrieve japanese children songs based on a sung query. The language model used a finite state automaton constructed from the lyrics of the songs in the test database, restricting word transitions to the next word in the lyrics or to the end symbol. Reported results were 75% correct word recognition. Authors of [3] constructed the language model and lexicon for each song from the lyrics, reporting over 70% correct recognition for most of the test songs.

In polyphonic music, the lyrics recognition problem becomes more difficult, and it relies on separating the vocals from the polyphonic mixture. Until now the use of speech recognition techniques for polyphonic music was limited to alignment of the text with singing, where the recognizer is given a sequence of phonemes or words and it only has to locate them within the acoustic segment [4, 5]. Authors of [4] use a speech recognizer adapted to singing voice to align lyrics with segregated vocals for japanese pop songs, with a language model containing the sequence of the vowels in the lyrics. Similar work was done for English language [5], with a language model containing the sequence of the words in the lyrics text.

An attempt of recognizing phonemes in polyphonic music was done in [6], using different classification techniques to classify individual samples of phonemes. The classification was not used to recognize words. To our knowledge there is no lyrics recognition system for the English language.

This paper studies the use of language models in the recognition of sung words and phonemes. The grammar for text-to-audio alignment [4, 5] uses the phonemes in a pre-established order. In the case of ”no-language model”, there are no constraints, any unit (word or phoneme) can follow any other, with no restrictions.

Between these two are the conventional n-gram language models which model the probabilities of short word sequences. We study the n-gram language models in representing phoneme and words sequences and apply them in singing phoneme/word recognition, using as test data clean singing and the vocal line separated from polyphonic audio.

The paper is organized as follows: Section 2 describes the singing recognizer: the phonetic HMM recognizer, language models and the algorithm used to separate the vocals from polyphonic music. Section 3 describes the acoustic material and the experimental results on phoneme and word recognition. Section 4 presents a query-by-singing application for songs retrieval based on the recognized words. Finally, in Section 5, the conclusions of the study are provided.

2. SINGING RECOGNITION

Speech and singing convey the same kind of semantic information and originate from the same production physiology. In singing, however, the intelligibility is often secondary to the intonation and musical qualities of the voice. Vowels are sustained much longer in

*This work was supported by the Academy of Finland.

singing than in speech and independent control of pitch and loudness over a large range is required. The dynamic range is greater in singing than in speech, and also the fundamental frequency variations of singing are of about 2 octaves for an average trained singer.

Still, speech and singing have many properties in common and it is plausible that singing recognition can be done using the standard technique in automatic speech recognition, a phonetic hidden Markov model (HMM) recognizer.

2.1. Phonetic HMM Recognizer

In HMM based speech recognition it is assumed that the observed sequence of speech vectors is generated by a hidden Markov model. An HMM consists of a number of states with associated observation probability distributions and a transition matrix defining transition probabilities between the states. We use an HMM speech recognizer consisting of 39 monophone models plus silence and short pause models, trained on a speech database [7]. The features for training the models are 13 mel-frequency cepstral coefficients plus delta and acceleration coefficients, calculated in 25 ms frames with a 10 ms hop between adjacent frames. Each phoneme is represented by a left-to-right HMM with 3 states. The silence model is a fully-connected HMM with 3 states and the short pause is a one-state HMM tied to the middle state of the silence model. The system was implemented using HTK [8].

Using two steps of constrained maximum likelihood linear regression (MLLR), the models were adapted to singing voice. In the first step, 8 regression classes were defined based on broad phonetic classes [9]: monophtongs, diphtongs, approximants, nasals, fricatives, plosives, affricates. In the second adaptation step, the base classes for adaptation were determined by clustering the mixture components based on acoustic similarities. The acoustic material used in adaptation is described in Section 3.1, and more details of the adaptation process are given in [10].

2.2. Language Model

A language model (LM) represents the linguistic restrictions present in the text/lyrics to be recognized, and acts by reducing the number of possible phonetic sequences from which the recognizer has to choose the most probable one. The language model comprises a vocabulary – a set of words that can be recognized by the system, and a set of rules that control the way these words can be combined into sentences.

The n-gram models are probabilistic models for predicting the next item in a sequence. For language modeling, these items can be phonemes, syllables, letters or words. In probabilistic terms, an n-gram language model provides probabilities $P(w_i|w_{i-1}, w_{i-2}, \ldots w_{i-n})$, meaning that it uses the previous $n-1$ words w_{i-1}, $w_{i-2}, \ldots,$ w_{i-n} to obtain the probability of the next word w_i [11]. If the number of occurences of the sequence of three words w_{i-2} w_{i-1} w_i and the sequence of two words w_{i-2} w_{i-1} are $C(w_{i-2}w_{i-1}w_i)$ and $C(w_{i-2}w_{i-1})$, then

$$P(w_i|w_{i-1}w_{i-2}) \approx \frac{C(w_{i-2}w_{i-1}w_i)}{C(w_{i-2}w_{i-1})}$$

The quality of a language model can be estimated by using it to compute a measure called perplexity on a previously unseen test set. Perplexity can be seen as the average size of the word set from which a word recognized by the system is chosen. The lower the number, the more accurately the language model is able to represent the text. An ideal language model should have small perplexity and small out-of-vocabulary (OOV) percentage (words not included in the vocabulary of the recognizer) on an unseen text.

2.3. Vocal Line Separation

The amount of interfering sounds in polyphonic music is high and therefore vocal separation is required prior the recognition. We apply the vocal separation algorithm proposed in [12], which uses pitch-based inference combined with background model subtraction. The method consists of the following processing steps.

The notes of the most prominent vocal melody line are estimated using a melody transcription algorithm. The algorithm is capable of discriminating between vocal and solo instrumental lines, at least to some degree. More accurate time-varying fundamental frequencies (f0s) are estimated by interpolating values of the fundamental frequency salience spectrogram. Based on the estimated f0s, a binary time-frequency spectrogram mask is generated. A time- and frequency-dependent background model is estimated using the non-vocal regions of the spectrogram. The background model is subtracted from the vocal regions of the spectrogram. Time-domain signal corresponding to the vocal regions is synthesized by using the phases of the original signal.

More detailed description of the processing steps is given in [12]. The method was found to produce relatively good separation quality and robustness on different genres of polyphonic music. Some amount of separation errors are inevitable because of undetected vocal notes or incorrectly estimated onsets or offsets, instrumental notes detected as vocal notes, or unresolved interference because of overlapping vocal and instrumental notes.

3. RECOGNITION EXPERIMENTS

Phoneme and word recognition experiments were conducted using different language models constructed for phonemes and words, on monophonic singing voice data and on vocal line separated from polyphonic music.

3.1. Acoustic Data

Monophonic singing was used both for testing the phoneme and word recognition and for adapting the HMMs to singing voice. This database, denoted vox_clean, contains 49 fragments of 12 popular songs, 19 male and 30 female pieces, with lengths ranging between 20 and 30 seconds. The adaptation and testing was done in a 5-fold setup, with one fifth of the data used as test set.

For testing the recognition on commercial music, we used 17 polyphonic songs. The songs were manually segmented into structurally significant units (chorus, verse), yielding 100 fragments with lengths between 9 and 40 seconds. We denote this database as $poly\_100$. For this testing case, the HMMs were adapted to singing voice using the entire vox_clean database. The lyrics of both databases were manually annotated.

3.2. Language Models

For phoneme recognition we constructed unigram, bigram and trigram language models. We assume that a phoneme language model is characteristic to the English language, and phoneme-level language models for speech data or lyrics cannot differ significantly. As training data for phoneme language models we used the text from the speech database that was used for training the acoustic models.

language model	training text	vox_clean	$poly\_100$
phoneme bigram	11.49	11.75	11.29
phoneme trigram	6.38	8.41	8.25
word bigram	90.20	147.08	97.80
word trigram	53.65	117.82	77.46
OOV %	5.90	2.21	2.50

Table 1. Perplexities of bigram and trigram phoneme and word language model on the training text (speech database transcriptions for phonemes, lyrics text for words) and on the test lyrics texts.

The database contains 1132 phonetically balanced sentences – over 48000 phoneme instances.

The perplexities of bigram and trigram phoneme language models ph_bg and ph_tg on the training text and on the lyrics text from vox_clean and $poly\_100$ databases are presented in Table 1. In the case of phoneme language model there is no concern over OOV, since all the phonemes are included in the vocabulary of the LM. According to the perplexities, the phoneme language model represents well the lyrics text, the perplexities being similar for all three texts.

A word language model for lyrics was constructed using the lyrics of 4470 songs containing over 1.2 million word instances, retrieved from www.azlyrics.com. From a total of approximately 26000 unique words, a vocabulary of 5167 words was chosen by keeping the words that appeared at least 5 times. Bigram and trigram models were constructed and their perplexities calculated on the training corpus and on the lyrics of vox_clean and $poly\_100$ databases. The perplexities are presented in Table 1 also. The percentage of OOV words on the training text represents mostly words in languages other than English, also the words that appeared too few times and were removed when choosing the vocabulary.

3.3. Evaluation

Phoneme and word recognition testing was done on monophonic singing and on vocal lines separated from polyphonic music. The percentage of correctly recognized phonemes or words and the accuracy of the recognition are defined in terms of the number of substitution errors S, deletion errors D and insertion errors I, reported for the total number of reference instances, N:

$$correct[\%] = [(N - D - S)/N] \times 100,$$

$$accuracy[\%] = [(N - D - S - I)/N] \times 100$$

The recognition rate and the accuracy differ by the rate of insertion errors. In the recognition process, the grammar scale factor and the word insertion penalty can be used to control the influence of the language model over the computed probabilities. The insertion penalty is a fixed cost added to each token when it transmits from the end of one word to the start of the next. The grammar scale factor is the amount by which the language model probability is scaled before being added to each token as it transits from the end of one word to the start of the next. These parameters influence the recognition performance and some experimental tuning is required to find optimal values for the task.

3.4. Phoneme and word recognition results

The values for the grammar scale factor and word insertion penalty are chosen experimentaly. Table 2 presents the average phoneme recognition results for the monophonic singing data, with the grammar scale factor $s = 5$ and insertion penalty $p = -10$, using no language model, unigram, bigram and trigram language models. For these values, the number of insertion and deletion errors for the bigram language model is nearly equal. The change brought by associating probabilities to phonemes by using the unigram language model affects both the correct percentage and accuracy of the recognition. The 2000 NIST evaluation of Switchboard corpus automatic speech recognition systems [13] reports error rates of 39-55% for phoneme recognition, while the lowest error rate ($100 - accuracy$) for clean singing recognition in Table 2 is approximately 70%.

	LM	Correct%	Accuracy%
clean singing	no LM	41.11	19.69
	ph_ug	36.20	27.50
	ph_bg	39.36	29.63
	ph_tg	46.05	22.54
polyphonic	no LM	23.45	5.83
	ph_ug	18.68	16.65
	ph_bg	21.79	19.16
	ph_tg	30.24	13.80

Table 2. Phoneme recognition for clean singing and vocal line extracted from polyphonic music, with no language model, unigram, bigram and trigram language models.

	LM	Correct%	Accuracy%
clean singing	word bigram	23.93	12.38
	word trigram	21.07	-1.38
polyphonic	word bigram	6.81	5.51
	word trigram	6.52	3.93

Table 3. Word recognition for clean singing and vocal line extracted from polyphonic music, with bigram and trigram language models.

Using the same grammar scale and insertion penalty values, with no further tuning for the trigram language model case, there is an important increase in the correct rate, with a decrease in accuracy. The ideal values for these parameters depend on the application. In speech recognition applications, the accuracy of the recognition is important, while thinking about information retrieval purposes even highly imperfect transcriptions of the lyrics can be useful.

Recognition results for the polyphonic music test data are also presented in Table 2. The separated vocal line is more difficult to recognize, because of some interference of other sources which have not been properly separated, and also artifacts caused by the separation algorithm. In some cases parts of the singing are missing – for example consonants being removed at the beginning of the word by the separation algorithm, resulting in recognition errors.

Word recognition of monophonic singing was tested in the 5-fold setting on the vox_clean database. The recognition results for bigram and trigram language models are presented in table 3. The best results obtained are the correct recognition of one fifth of the words, using the bigram language model. Recognition rate of singing extracted from polyphonic music using the same vocabulary and language models is also presented in the same table.

correct transcription	recognized text
cause it's a bittersweet	cause I said bittersweet
symphony this life	symphony this our life
trying to make ends meet	trying to maintain sweetest
you're a slave to the money	ain't gettin' money
then you die	then you down

Table 4. Examples of errors in recognition.

	no. of rec. songs	recognized [%]
Top 1	28	57%
Top 5	33	67%
Top 10	35	71%

Table 5. Query-by-singing retrieval accuracy

4. QUERY-BY-SINGING BASED ON WORD RECOGNITION

In query-by-humming/singing, the aim is to identify a piece of music from its melody and lyrics. In a query-by-humming application, the search algorithm will transcribe the melody sung by the user and will try to find a match of the sung query with a melody from the database. For large databases, the search time can be significantly long. Assuming that we also have the lyrics of the songs we are searching through, the words output from a phonetic recognizer can be searched for in the lyrics text files. This will provide additional information and narrow down the melody search space. Lyrics will be more reliable in the case of less skilled singers.

The output of the recognition system offers sometimes words that are acoustically very similar with the correct ones, sometimes cases with different spelling but same phonetic transcription. For recognition performance evaluation they count as errors, but for music information retrieval purpose, we do not need perfect transcription of the lyrics. Some examples representing typical recognition results can be found in Table 4.

We built a small retrieval system based on sung queries recognized by the system presented in Table 3 (23.93% correct recognition rate), that uses a bigram language model to recognize the clean singing voice in the presented 5-fold experiment. For this purpose, we constructed a lyrics database consisting of the text lyrics of the fragments that form the $poly\_100$ and vox_clean databases, using as queries the 49 singing fragments of the vox_clean database.

For retrieval we use a bag-of-words approach, simply searching for each recognized word in all the lyrics text files and ranking the lyrics files according to the number of matched words. We consider a song being correctly identified when the queried fragment appears among the first N ranked lyrics files. Table 5 presents the retrieval accuracy for N being 1, 5 and 10. The application shows promising results, the first retrieved song being correct in 57% of the cases.

5. CONCLUSIONS

We have studied the use of n-gram language models in recognizing phonemes and words in monophonic and polyphonic music. Phoneme-level language models were trained on speech, but the perplexities on textual lyrics indicate that the models reliably represent the language constrains in lyrics. Word-level language models were trained on textual lyrics for a vocabulary of 5167 words. The highest recognition rates for monophonic and polyphonic singing were obtained with the trigram language model, with higher accuracy for the bigram language model when using the same grammar factor and insertion penalty.

A text-based song retrieval application using the recognized words correctly retrieves the queried song for over half of the fragments in our database. The first retrieved song is correct in 57% of cases; moreover, the correct song is in the first 10 ranked songs in the database in over 70% of cases. Even with such low word recognition rates this proves that a textual query-by-singing can be useful for narrowing the melody search.

References

[1] F. Wiering R. Typke and R. C. Veltkamp, "Mirex symbolic melodic similarity and query by singing/humming," in *Intl. Music Information Retrieval Systems Evaluation Laboratory(IMIRSEL)*, 2006.

[2] A. Ito M. Suzuki, T. Hosoya and S. Makino, "Music information retrieval from a singing voice using lyrics and melody information," *EURASIP Journal on Advances in Signal Processing*, 2007.

[3] S. Hayamizu A. Sasou, M. Goto and K. Tanaka, "An autoregressive, non-stationary excited signal parameter estimation method and an evaluation of a singing-voice recognition," in *Proc. of the 2005 IEEE Intl. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, Philadelphia, USA, 2005.

[4] H. Fujihara, M. Goto, J. Ogata, K. Komatani, T. Ogata, and H. G. Okuno, "Automatic synchronization between lyrics and music CD recordings based on Viterbi alignment of segregated vocal signals," in *ISM '06: Proc. of the Eighth IEEE Intl. Symposium on Multimedia*, Washington, DC, USA, 2006.

[5] A. Mesaros and T. Virtanen, "Automatic alignment of music audio and lyrics," in *Proc. of the 11th Intl. Conference on Digital Audio Effects (DAFx-08)*, Helsinki, Finland, 2008.

[6] K. Schmidt M. Gruhne and C. Dittmar, "Phoneme recognition in popular music," in *Proc. of the Intl. Symposium on Music Information Retrieval*, Vienna, Austria, 2007.

[7] "CMU ARCTIC databases for speech synthesis," http://festvox.org/cmu_arctic/.

[8] "Cambridge University Engineering Department. The Hidden Markov Model Toolkit (HTK)," http://htk.eng.cam.ac.uk/.

[9] C. J. Leggeter and P. C. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density hidden markov models," *Computer Speech and Language, No. 9*, 1995.

[10] A. Mesaros and T. Virtanen, "Adaptation of a speech recognizer to singing voice," in *Proc. of 17-th European Signal Processing Conference (EUSIPCO-2009)*, Glasgow, Scotland, 2009.

[11] D. Jurafsky and J. H. Martin, *Speech and language processing*, Prentice Hall, Upper Saddle River, New Jersey, 2000.

[12] T. Virtanen, A. Mesaros, and M. Ryynänen, "Combining pitch-based inference and non-negative spectrogram factorization in separating vocals from polyphonic music," in *ISCA Tutorial and Research Workshop on Statistical And Perceptual Audition SAPA*, Brisbane, Australia, 2008.

[13] J. Hollenback S. Greenberg, S. Chang, "An introduction to the diagnostic evaluation of the switchboard-corpus automatic speech recognition systems," in *Proc. of the NIST Speech Transcription Workshop*, College Park, MD, 2000.