EURASIP Journal on
Audio, Speech, and Music Processing
a SpringerOpen Journal

RESEARCH Open Access

Exploiting contextual information for prosodic event detection using auto-context

Junhong Zhao[1,2*], Wei-Qiang Zhang[3], Hua Yuan[3], Michael T Johnson[4], Jia Liu[3] and Shanhong Xia[1]

Abstract

Prosody and prosodic boundaries carry significant information regarding linguistics and paralinguistics and are important aspects of speech. In the field of prosodic event detection, many local acoustic features have been investigated; however, contextual information has not yet been thoroughly exploited. The most difficult aspect of this lies in learning the long-distance contextual dependencies effectively and efficiently. To address this problem, we introduce the use of an algorithm called auto-context. In this algorithm, a classifier is first trained based on a set of local acoustic features, after which the generated probabilities are used along with the local features as contextual information to train new classifiers. By iteratively using updated probabilities as the contextual information, the algorithm can accurately model contextual dependencies and improve classification ability. The advantages of this method include its flexible structure and the ability of capturing contextual relationships. When using the auto-context algorithm based on support vector machine, we can improve the detection accuracy by about 3% and F-score by more than 7% on both two-way and four-way pitch accent detections in combination with the acoustic context. For boundary detection, the accuracy improvement is about 1% and the F-score improvement reaches 12%. The new algorithm outperforms conditional random fields, especially on boundary detection in terms of F-score. It also outperforms an n-gram language model on the task of pitch accent detection.

Keywords: Prosodic event detection; Auto-context; Pitch accent; Boundary; Support vector machines

1 Introduction

Speech is often characterized across two levels of expression: the segmental level encompassing basic phonetic meaning and the prosodic level with additional suprasegmental information. The prosodic level expression plays a crucial role in speech communication, carrying much linguistic and paralinguistic information. Prosody enables listeners to recover word meanings, emphasis, and speaker intent and attitude. In addition, prosody carries information about the speaker's emotional state. Prosody primarily manifests itself as pitch accent, pause, variations in speaking rate, and intonation. These prosodic events are realized by modulating acoustical correlates such as duration, pitch, and intensity at a syllable, word, or whole utterance level.

In spoken language processing, the detection of prosodic events is a primitive step needed for computers to access the critical high-level information regarding human speech interaction. As such, this task has wide application. It can provide assistance for automatic prosody annotation. Since the manual annotation of prosody for speech synthesis or speech understanding is time-consuming and laborious, the detection of prosodic events can provide substantial time savings. For second language learning, there is potential for computer-assisted language learning (CALL) systems to incorporate prosodic event detection to help detect learner mispronunciations at a prosodic level and provide feedback for improving pronunciation naturalness. Prosodic event detection can also be used as foundation for the downstream spoken language processing tasks such as speech summarization and topic segmentation.

Due to the suprasegmental nature of prosody, contextual information is very important for prosodic event recognition. Here, 'context' refers to the correlation between each prosodic unit and its surroundings. There

*Correspondence: junhong.iecas@gmail.com
[1]State Key Laboratory of Transducer Technology, Institute of Electronics, Chinese Academy of Sciences, No.19 North west Rd., Beijing 100190, China
[2]University of Chinese Academy of Sciences, No. 19A Yuquan Road, Beijing 100190, China
Full list of author information is available at the end of the article

are several types contextual information, including nearby acoustic appearances ('acoustic context'), nearby prosodic event distributions ('prosodic context'), and nearby lexical and syntactic appearances ('textual context'). In this paper, we focus only on acoustic context and prosodic context, but do not address linguistic effects. Perceptually, pitch accent is perceived when the related acoustic features of a syllable stand out from its surroundings. In addition to the influence of the adjacent syllables, the overall surface realization of accent is also affected by many other broader-scale phenomena such as the presence of phrasal boundary, phrase structure, and topic [1,2].

In this paper, we investigate the utilization of contextual information for pitch accent and boundary detection by using the *auto-context* algorithm, which was first proposed in [3] for high-level computer vision tasks like image segmentation. In this algorithm, the classification probabilities obtained from the preceding iteration are used to provide possible contextual clues, together with acoustic features to improve the next iteration. Each detection object is supported by combinations of contextual probabilities from any contextual range. Our experimental results show that this algorithm enhances detection performance for both pitch accent and boundary detection tasks.

2 Related work

Many approaches have been explored for prosodic event detection. The target unit used to indicate the prosodic events can be a phone, syllable or word, with acoustic features typically modeled by statistical machine learning methods. In [4], Conkie et al. used short-frame speaker-normalized pitch and energy as acoustic representations modeled by prosodic context-dependent HMMs. In [5], Ananthakrishnan and Narayanan modeled the frame-level acoustic features using coupled hidden Markov models (CHMMs) to detect pitch accent and boundary in binary modes.

This work assumed that the modulation of prosody-related acoustic parameters, such as increase of the local energy, extension of the duration, and exaggeration of pitch movements, were all asynchronous and could be modeled by CHMMs with multiple data streams. In [6], bagging and boosting ensemble machine learning methods were adopted based on a decision tree learning algorithm. Both methods improved the overall accuracy of four-category pitch accent classification.

The work in [7] utilized a decision tree to model acoustic features. Using the posterior probabilities provided by the decision tree, a bigram prosodic label sequence model was combined to detect pitch accent and boundary tones at the syllable level. In [8], Ananthakrishnan and Narayanan used a maximum *a posteriori* (MAP) framework with multiple classifiers including GMMs, linear decision discriminants, and neural networks (NNs) to detect pitch accent and boundary, with NNs giving the best performance. When combined with 4-gram delexicalized language model, this method achieved an accuracy of 80.1% on binary pitch accent classification and 89.6% on boundary detection. Jeon and Liu used NNs [9] and support vector machines (SVMs) [10] for acoustic modeling. With NNs, pitch accent and boundary detection accuracy reached 83.5% and 84.8%, respectively. The performance of the SVM classifier was better than that of NNs, achieving 85.7% accuracy for pitch accent detection. Recently, conditional random fields (CRFs) have become popular in prosodic event detection. In [11], acoustic features were modeled by a linear-chain CRF as well as a two-level factorial CRF. Linear-chain CRF has been used extensively in recent work [2,12,13]. These reference approaches to prosodic event detection have been summarized in Table 1.

To investigate the importance of contextual information in prosodic event detection, the work in [16] examined the detection performance of pitch accent at word, syllable, and vowel levels, respectively. When using a constant amount of context, the results showed that detection in the word domain achieved the best performance, showing that acoustic excursions exist beyond syllable range. In [1], the contextual influence of local coarticulatory constraints and broader range phrasal effects were investigated for the detection of prominence. The results showed that the incorporation of local acoustic context can significantly improve the detection performance, with phrasal effects less significant.

With respect to utilizing contextual information in acoustic modeling, there are two representative methods: an n-gram language model [7,8] and CRF [2,11-13] model. An n-gram language model assumes that the current prosodic state is dependent on its finite histories, with dependencies established in the form of conditional probability. CRFs are a class of graphical models that are undirected and conditionally trained. For the commonly used linear chain CRF, the dependencies between prosodic labels are modeled in a pairwise neighborhood structure. CRFs have the advantage of modeling the relationships between sequential labels and have been proven efficient in prosodic prediction [2,11-13]. Figure 1 shows the dependency diagrams of the two models. Unlike these models, auto-context simultaneously integrates the acoustic features together with the context information by learning a series of classifiers. As discussed in [3], auto-context can integrate any mode of neighborhood structure, including long range, to make good use of contextual information. It is up to the learning algorithm to select and fuse the informative context and acoustic features.

Table 1 Summary of different approaches for prosodic event detection

Reference	Corpus	Domain	Model	Class	Pitch accent accuracy (%)	Boundary accuracy (%)
[4]	TTS and BN	Word	Prosodic context-dependent HMM	Binary	82.8	-
[5]	BURSC	Word, syllable	CHMM	Binary	72.0 74.0	77.3 86.0
[14]	BURSC	Syllable	GMM	Binary	77.3	68.2
[7]	BURSC	Syllable	Decision tree and n-gram language model	Binary	84.0	71.0
[8]	BURSC	Syllable	NN and n-gram language model	Binary	80.1	89.6
[9]	BURSC	Syllable	NN	Binary	83.5	84.8
[10]	BURSC	Syllable	SVM	Binary	85.7	-
[6]	BURSC	Syllable	Adaboost CART	Four	84.7	-
[15]	BURSC	Word	MaxEnt	Binary	80.1	82.7
[11]	BURSC	Syllable	CRF	Binary, four	79.5 77.1	92.4

3 Corpus and tasks

The data corpus used in this work is the Boston University Radio Speech Corpus (BURSC) [17], a standard corpus for prosody event detection and prediction studies. This corpus is composed of news stories read by seven FM radio news announcers. Each paragraphed-size utterance typically consists of several sentences and is hand-annotated with the orthographic transcription, phonetic alignments, part-of-speech (POS) tags, and prosodic labels based on ToBI conventions. Utterances from two females (F1, F2) and two males (M1, M2) constitute the training and testing dataset used here. The distribution of these utterances is listed in Table 2.

There are four parallel tiers for ToBI annotation conventions to describe the prosodic events [18]. Among these, the tone tier annotates the presence of pitch accent (* suffix) and phrase boundaries. There are two basic types of accent, high (H) and low (L), which can be further divided into subclasses such as downstepped accent (! prefix). The phrase boundaries include intermediate phrase boundary (- suffix) and intonational phrase boundary (% suffix), which follow the different types of phrase accent and boundary tones. The break tier is used to describe the disjuncture between words. The degree of disjuncture from weak to strong is marked by break indices ranging from 0 to 4. The phrase boundary locations usually score 3 or above, where '3' indicates an intermediate phrase boundary and '4' indicates a full intonational phrase boundary. These two kinds of boundaries are different in their degree of salience. Figure 2 shows a ToBI annotation example for the phrase 'design improvement and schedule'. The top three layers are the orthographic tier, the tone tier, and the break tier, respectively.

In our work, we have implemented syllable partitions and take the syllable as the domain of pitch accent and boundary detection. Using the detailed representation of prosodic types from the ToBI annotation framework can

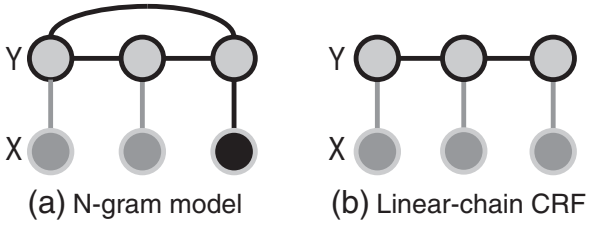

Figure 1 The relational structures adopted by n-gram and CRF models. Here, X represents the features, and Y represents the labels which are being predicted. **(a)** N-gram model. **(b)** Linear-chain CRF.

Table 2 The distribution of the dataset used in our experiment

	F1	F2	M1	M2
Number of utterances	74	166	72	51
Number of sentences	279	1,176	391	209
Number of words	3,993	12,060	5,059	3,608
Number of syllables	6,580	20,836	8,168	5,915
Number of accents	2,253	7,063	2,564	1,933
Number of boundaries	977	3,702	1,092	882
Accent occupancy (%)	34.3	33.9	31.4	32.7
Boundary occupancy (%)	14.9	17.8	13.4	14.9

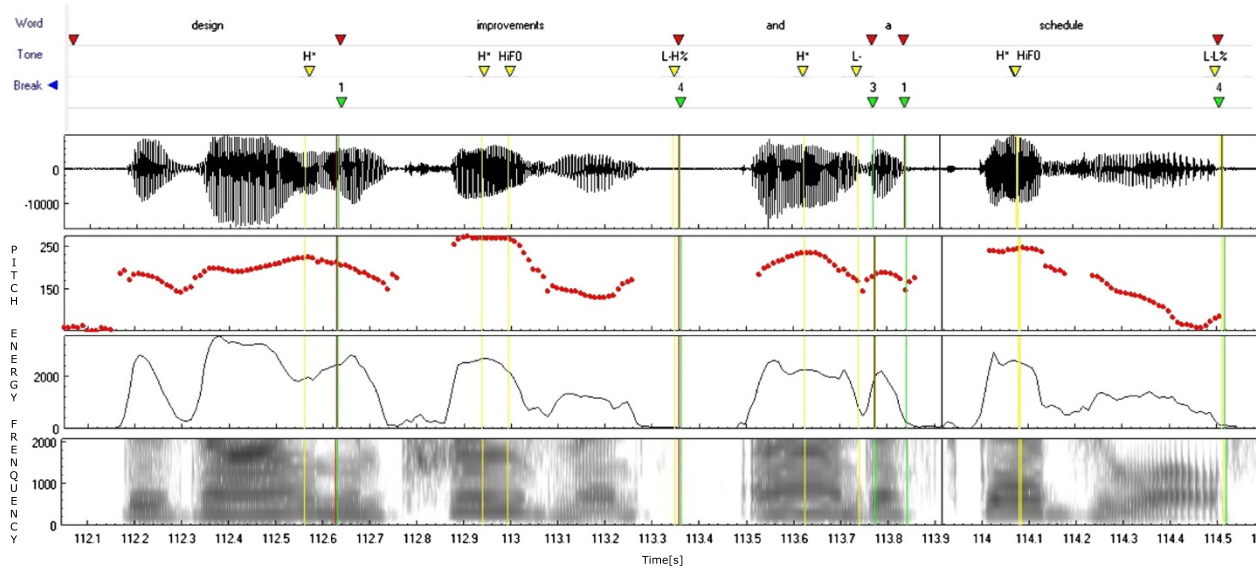

Figure 2 An example of ToBI annotation for the phrase 'design improvement and schedule'. The example includes the acoustic appearances of pitch, energy, and frequency spectrum.

cause serious data sparsity problems since there are only a few examples for some prosodic types. Considering a balance between the amount of the training data and detection fineness, in this work, we implement pitch accent detection tasks in two-way and four-way modes, following previous approaches [8,9,11,13]. For the two-way task, we divide the syllables into accented and unaccented based on the presence of an asterisk mark. As Table 2 shows, in this classification case, the percentage of the accented syllables is about 33%. For the four-way task, we decompose the pitch accent into three types: high, low, and downstepped, in addition to the unaccented type. Four-way classification increases the detection complexity and fineness, but causes some data sparsity. For boundary detection, we set our detection task as a binary (presence/absence) classification problem, which identifies whether the syllable is followed by a boundary or not. Although intonational phrase boundaries can be more reliably detected and have been widely applied in many downstream spoken language processing tasks such as speech summarization, intermediate phrase boundaries are also important elements for phrasal analysis and prosodic annotation. Here, we treat the intonational and intermediate phrase boundaries equally and group the break indices of '3' and '4' together to represent our 'boundary' category, as has been done in some previous works [8,13,19]. As shown in Table 2, the syllables with boundary presence represent about 15% of the dataset. The boundary detection results obtained in this way can be used directly or as preliminary information that can be followed by a further identification of the boundary salience at the presence positions. We summarize the clusters of ToBI labels and their mapping relationships with prosodic categories for detection in Table 3.

4 Prosodic event detection method: auto-context

In this paper, the auto-context algorithm is introduced for prosodic event detection. The basic objective of the auto-context algorithm is to maximize $p(y_i|\boldsymbol{X})$ for all samples, where $\boldsymbol{X} = (x_1, \cdots, x_n)$ is the input feature vector, and y_i is the class label for sample i. The auto-context algorithm provides an iterative way to asymptotically approach this objective as follows: Given a set of training samples with ground truth labels $S = \{(Y_i, \boldsymbol{X}(i)), i = 1, \ldots, m\}$, a classifier is first trained using local features so the probabilities of classes for each sample can be obtained. In the ensuing iterative process, the probabilities of both the current and surrounding samples obtained in the current iteration t are combined together as a probability vector $\boldsymbol{P}^t(i)$. This is then concatenated with the original acoustic features to construct a new feature vector. The new feature vectors of all the samples compose the training set $S_{t+1} = \{Y_i, [\boldsymbol{X}(i), \boldsymbol{P}^t(i)]\}$ for iteration $t + 1$. Using this updated training set, the new model is trained.

During the iterative process, the auto-context algorithm selects the informative contexts automatically and fuses them with appearance cues. At first, samples with strong discriminant cues will be correctly classified by the initial model and obtain stable posterior probabilities. These probabilities can then influence their neighbors in subsequent iterations, especially when there are close correlations between them. Convergence has been proven,

Table 3 Mapping between clusters of ToBI labels and prosodic categories for detection

Annotation tier	ToBI labels	Four-way pitch accent	Two-way pitch accent	Boundary
Tone tier	H*, L+H*, *, *?, X*?	High	Accent	-
	L*, L*+!H, L*+H	Low		
	!H*, H+!H*, L+!H*	Downstepped		
	Others	Unaccent		
Boundary tier	3, 4	-		Boundary presence
	Others	-		Boundary absence

with a monotonically decreasing training error, as shown in [3].

We adopt this algorithm to explore contextual information for pitch accent and boundary detection. We choose syllables from one to five syllables away from the central syllable as contextual regions to conduct our analysis. In our tasks, the posterior probabilities $\boldsymbol{P}(i)$ are used as the prosodic context (as opposed to 'acoustic context') since they present the likelihoods of syllables belonging to different prosodic events. Acoustic features can include not only local features but also acoustic context. Correspondingly, $\boldsymbol{X}(i)$ can be decomposed as the local features vector $\boldsymbol{X}_{\text{local}}(i)$ and acoustic context vector $\boldsymbol{X}_{\text{context}}(i)$, as Eq. (1) denoted. Here, we will consider both of them. The local acoustic feature $\boldsymbol{X}_{local}(i)$ used in this work will be described in detail in Section 5.1. The first-order differential values are used as the acoustic context $\boldsymbol{X}_{\text{context}}(i)$. We use the following steps to implement the algorithm:

1. The acoustic features, including local features $\boldsymbol{X}_{\text{local}}(i)$ and acoustic context $\boldsymbol{X}_{\text{context}}(i)$, are used to train the initial acoustic model. After the first round of training and testing, we obtain the class probabilities.
$$\boldsymbol{X}(i) = [\boldsymbol{X}_{\text{local}}(i), \boldsymbol{X}_{\text{context}}(i)], \tag{1}$$
2. The contextual information of each syllable is incorporated by combining the probabilities from its neighbors for the next iteration. The probability vector $\boldsymbol{P}(i)$ for the ith sample is constructed as follows:
$$\boldsymbol{P}(i) = [\boldsymbol{C}_*(i-n), \cdots, \boldsymbol{C}_*(i-1), \boldsymbol{C}_*(i), \boldsymbol{C}_*(i+1), \cdots, \boldsymbol{C}_*(i+n)], \tag{2}$$
where $\boldsymbol{C}_*(i)$ represents any collection of classification probabilities for the ith syllable, and n is the range parameter controlling the extent of the context. After extension, the new training set for the second stage becomes
$$S(i) = \{y_i, [\boldsymbol{X}(i), \boldsymbol{P}(i)], i = 1, 2, 3, \ldots, m\} \tag{3}$$
3. Using the $S(i)$ training set, we train the next acoustic model and update $\boldsymbol{P}(i)$.
4. Steps 2 and 3 are repeated until convergence to let the algorithm recursively learn the informative pitch accent context automatically.

The flowchart of the algorithm is depicted in Figure 3.

5 Acoustic representation

5.1 Features for pitch accent

Pitch accent typically correlates with a higher level of pitch and energy and an increased duration. We extracted acoustic features based on these acoustic measurements in syllable regions to detect pitch accent. These features can be categorized into two groups: frame-averaged features, including the mean values of pitch, energy and duration; and TILT features provided by parameterizing the pitch contour using the TILT model. In addition, the forward and backward difference values for both frame-averaged features and TILT features are extracted as acoustic contextual information. In the following section, we describe how to extract these features in detail.

5.1.1 Frame-averaged features

Loudness Pitch accent is closely related to human auditory characteristics. For perceptual accuracy, loudness can be used instead of intensity to detect pitch accent [20]. Here, we use a loudness model proposed by Zwicker and Fastl [21] to extract the loudness feature. This starts with using the short-time Fourier transform (STFT) to transfer the signal from the temporal to the frequency domain. The linear-scale frequency is grouped into a critical band rate scale to model the human hearing mechanism. This mapping relationship is given by

$$z(\text{Bark}) = T(f) = 13\tan^{-1}(0.00076f) + 3.5\tan^{-1}\left(\frac{f}{7500}\right)^2, \tag{4}$$

where f denotes frequency in Hertz, z represents the critical band rate measured in Bark units, and $T(\cdot)$ is the transform function between them. Here, we divide the audible range into 24 critical bands. The intensity of each critical band is obtained by summing up all the frequency points that are distributed within the band range of

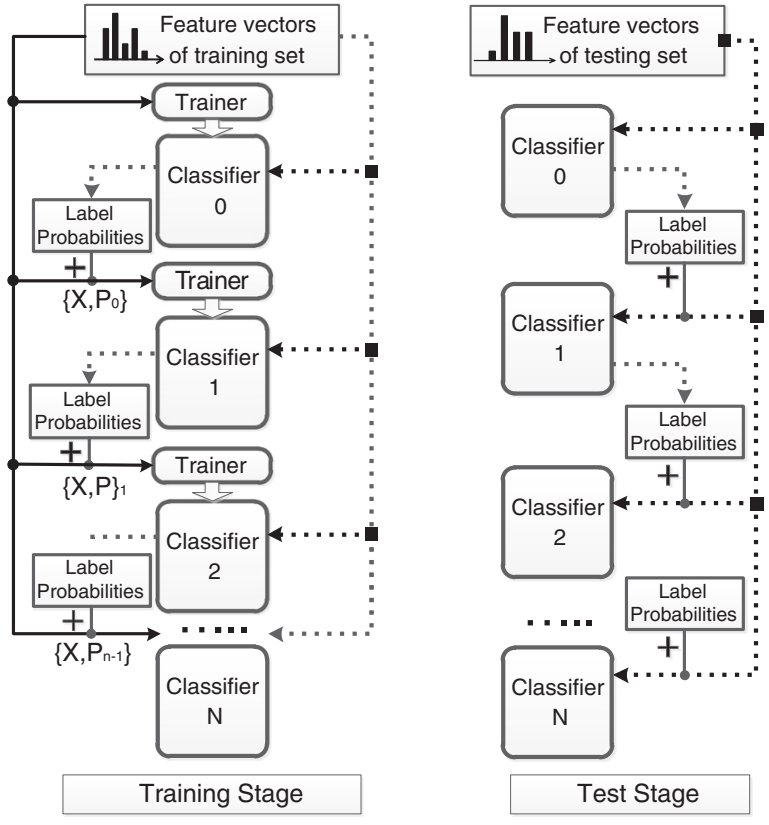

Figure 3 Flowchart of the auto-context algorithm.

$(z-0.5, z+0.5)$ and then by calculating the corresponding sound pressure level (SPL) according to

$$I(z) = 10\log\left(\frac{\int_{L(z)}^{H(z)} I(f)df}{I_0}\right) \text{ dB},$$
$$\text{where } H(z) = T^{-1}(z+0.5),\ L(z) = T^{-1}(z-0.5). \tag{5}$$

Here, $I_0 = 10^{-12}\ W/m^2$ is the standard threshold of hearing at 1 kHz.

Stevens pointed out that the relationship between intensity and perceptual loudness obeys the power law [22]. Following this law, we calculate the loudness of per Bark based on $I(z)$ by

$$L(z) = 0.08\left(\frac{I_Q(z)}{I_0}\right)^{0.23}\left[\left(0.5+0.5\frac{I(z)}{I_Q(z)}\right)^{0.23}-1\right], \tag{6}$$

where $L(z)$ denotes the specific loudness in the zth Bark, and $I_Q(z)$ is the correponding threshold of intensity in quiet enviroment. The total loudness L of the frame is given by the summation of every critical band loudness $L = \sum_{z=1}^{24} L(z)$.

Semitone Based on a similar perceptual consideration, we transform the pitch values in Hertz to the semitone scale to better match with human perception [23]. Raw pitch values are calculated first using Praat [24], then a log-scale transformation is conducted according to the following equation:

$$S = 69 + 12\log_2\left(\frac{f}{440}\right), \tag{7}$$

where f is the fundamental frequency in Hz, and S is in semitones.

Spectral emphasis Previous studies have shown that midfrequency energy is more effective in accent classification than full-range distributed energy [25]. Midfrequency refers to the frequencies between 500 and 2,000 Hz. In this work, we use a finite impulse response (FIR) filter with Kaiser window to extract the energy within this bandwidth as a spectral emphasis feature.

Duration We compute the syllable duration using the boundary information generated by forced alignment. The speaker-independent speech recognizer was trained using the data from BURSC and the corresponding manual transcriptions.

Among these four features, the loudness, spectral emphasis, and semitone values are all extracted at a frame-by-frame level in the first iteration. After this, the loudness and the spectral emphasis are averaged across a syllable scope, while the semitone is averaged across a syllable nucleus to obtain frame-averaged values. In order to reduce the negative impact caused by different speakers and speaking rates, these features are all normalized by the mean value across the whole sentence.

5.1.2 TILT features

We follow the rise/fall/connection (RFC) model proposed in [26] to extract the TILT parameters as the representation of the pitch variation [27]. As an intonation model, the RFC model first categorizes the F0 contour into rise (R), fall (F), and connection (C) cases according to pitch trend and then continuously parameterizes the contour based on this classification. Here, we still use the semitone-scaled F0 contour of each syllable nuclei to extract the TILT features.

Linear interpolation is implemented to smoothen the contour. Then, a labeling procedure is conducted to mark the contours with one of the three kinds of categories. Labels of successive frames are merged together to be a single interval. Within the range of these divided areas, the amplitude-related measurement (tilt_a) is calculated by

$$\text{tilt}_a = \frac{|A_{\text{rise}}| - |A_{\text{fall}}|}{|A_{\text{rise}}| + |A_{\text{fall}}|}, \tag{8}$$

the duration-related measurement (tilt_d) is calculated by

$$\text{tilt}_d = \frac{D_{\text{rise}} - D_{\text{fall}}}{D_{\text{rise}} + D_{\text{fall}}}, \tag{9}$$

and the overall measurement of tilt (tilt_t) is calculated by

$$\text{tilt}_t = \frac{|A_{\text{rise}}| - |A_{\text{fall}}|}{2\,(|A_{\text{rise}}| + |A_{\text{fall}}|)} + \frac{D_{\text{rise}} - D_{\text{fall}}}{2\,(D_{\text{rise}} + D_{\text{fall}})}. \tag{10}$$

A_{rise} and A_{fall} are the sum of the rise and fall amplitudes, respectively, and D_{rise} and D_{fall} represent the sum of the rise and fall durations, respectively. We extracted these features using the Edinburgh Speech Tools Library (EST). In addition, we also included the maximum semitone that can be calculated directly by EST as one of the tilt features.

5.2 Features for boundary

As discussed in [28], the presence of a phrase boundary typically correlates with the presence of silence, the reset of pitch and energy, and the lengthening of pre-boundary duration. Each of these play a role in perception of increased disjuncture. We assume that acoustic variations caused by boundary phenomenon exist only within the region of syllables, and extract acoustic features for boundary detection using a syllable unit, as has been done in [7,8,19]. We assume that the region affected by the boundary covers the end of the pre-syllable, the silence interval (possibly absent), and the beginning of the post-syllable, while the candidate result of this region is assigned to the pre-syllable. The acoustic features include the acoustic measurement of the preceding and following syllables and their differential features across syllable boundaries for boundary detection. There are 25 feature dimensions in total, as follows:

1. The duration of the two syllables and their ratio value (3)
2. The duration of the two syllable nuclei and their ratio values (3)
3. The silence duration between the two syllables (1)
4. The means and maxima of pitch of the two syllables, and their differential values (6)
5. The loudness and spectral emphasis mean of the two syllables, and their differential values (6)
6. The amplitude, duration, and overall measurements of the TILT features of the two syllables (6).

Although one can describe a linguistic grammar that only word-final syllables can contain a boundary, we choose to take no linguistic constraints into the implementation of our algorithm and allow all possibilities for all syllables. We just let the algorithm itself to make decisions according to acoustic features and contextual information.

6 Experiments

We conduct a number of prosodic event detection experiments in this section. First, a classifier selection is conducted. The performances of the different classifiers are investigated, and the one with best performance is chosen as the baseline classifier for the auto-context experiments. The auto-context algorithm is then implemented for two-way and four-way pitch accent detections and for boundary detection. In these experiments, the effectiveness of the auto-context approach is verified from different aspects. Finally, comparisons are made between the auto-context, CRF, and n-gram methods.

The BURSC F1, F2, M1, and M2 data described in Section 3 is used for experimental evaluation. We use randomly selected fivefold cross-validation for each experiment. Accuracy and F-score are utilized to measure the performance of the pitch accent detection and boundary detection tasks. In addition, for boundary detection, we investigate the syllable-level detection performance, in which the measurement is presented as a fraction of all syllables, as well as the word-level detection performance, in which the measurement is presented as a fraction of word-final syllables. All test results presented here are obtained by averaging over the five cross-validation test

Table 4 Performance of prosodic event detection by NN and SVM classifiers using all combined features

	Two-way (%)		Four-way (%)		SL boundary (%)		WL boundary (%)	
	Accuracy	F-score	Accuracy	F-score	Accuracy	F-score	Accuracy	F-score
NN	80.5	66.8	75.0	58.9	87.7	43.7	80.3	44.1
SVM	81.2	70.8	76.3	59.9	88.1	45.3	81.2	46.2

Here 'Two-way' and 'Four-way' mean the pitch accent detection tasks in two-way and four-way modes, respectively, and 'SL boundary' and 'WL boundary' mean the boundary detection tasks measured at the syllable-level and word-level, respectively.

folds. Acoustic model parameters for both baseline and proposed methods are optimized using a development dataset. This development dataset is constructed from the F3 and M3 data of the BURSC corpus, including 57 utterances, 282 sentences, and about 8,000 syllables. All experiments are implemented by first optimizing the methods using the development data and then testing with fivefold cross-validation over the primary dataset.

6.1 Classifier selection

As mentioned in Section 4, the auto-context algorithm produces a sequence of classifiers through an iterative process. The performance of the chosen classifier will play a vital role in determining the final performance. The auto-context algorithm is not dependent on any specific classifier and can use either SVM or NN approaches. Since each of these have been used extensively in prosodic event

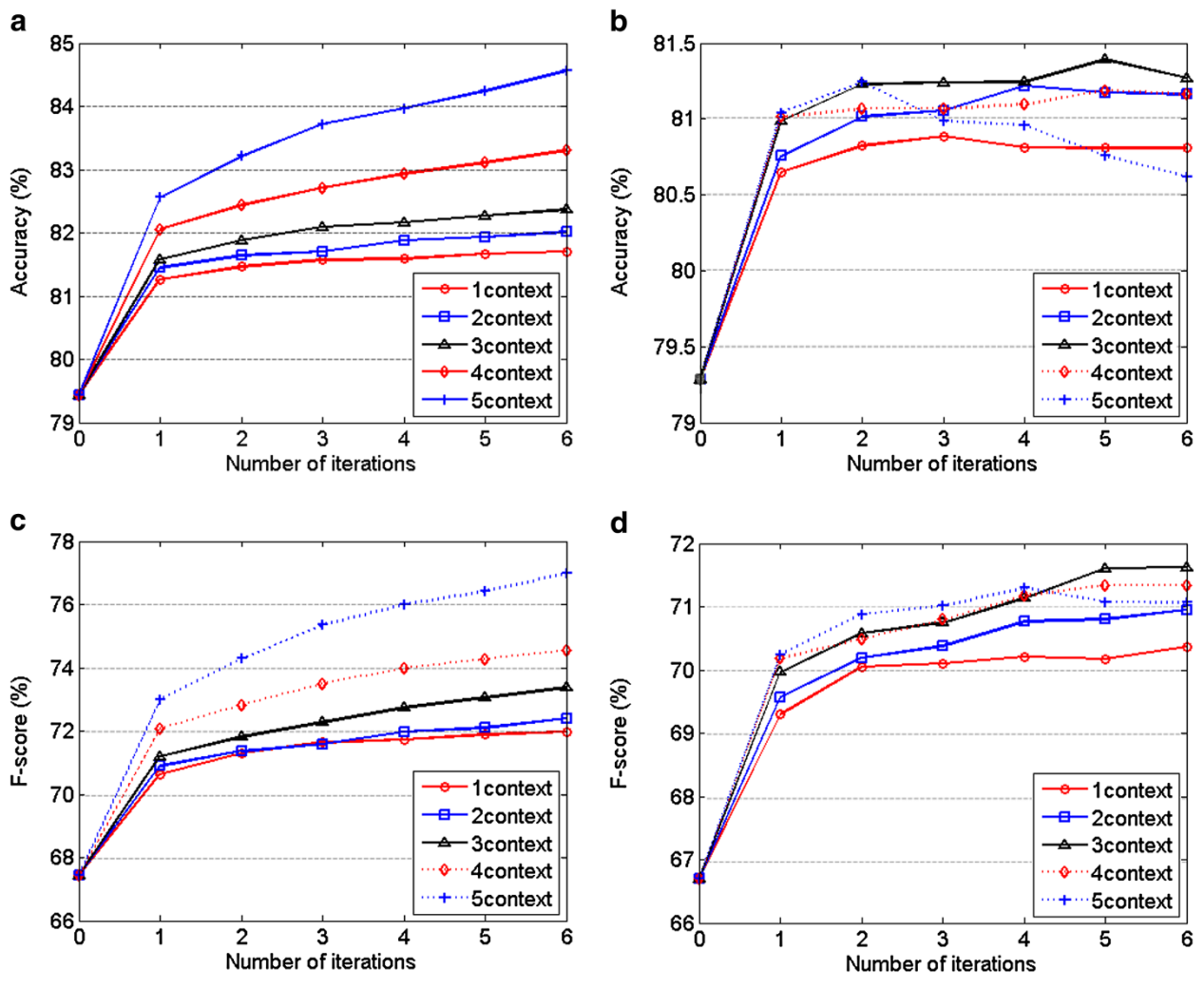

Figure 4 Performance of the auto-context algorithm for two-way pitch accent detection without acoustic features. (a) Accuracy performance on training data. **(b)** Accuracy performance on test data. **(c)** F-score performance on training data. **(d)** F-score performance on test data.

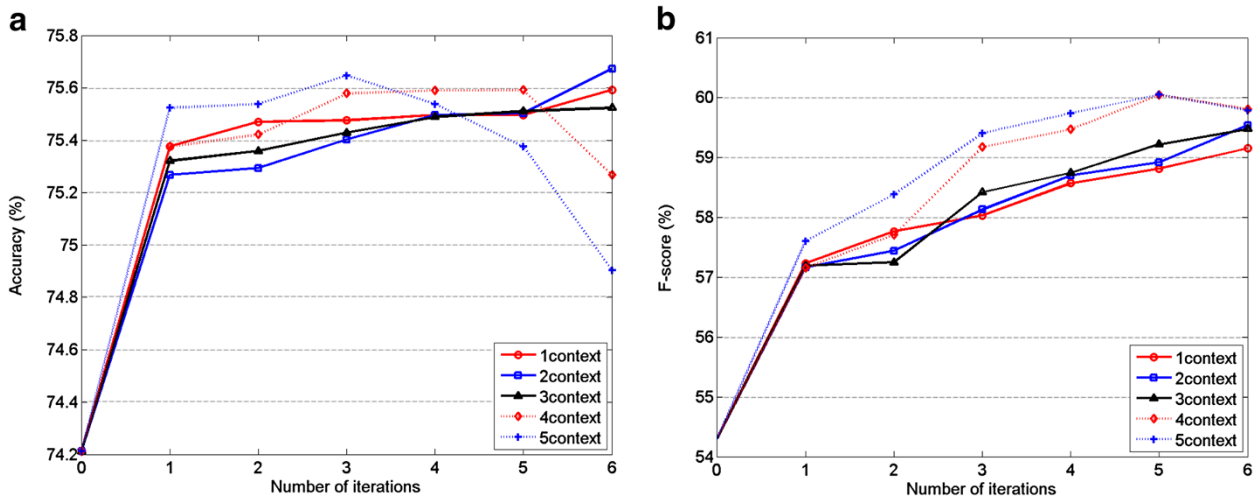

Figure 5 Performance of the auto-context algorithm on test dataset for four-way pitch accent detection without acoustic features. **(a)** Accuracy performance on test data. **(b)** F-score performance on test data.

detection, we will first conduct the investigation for their performance here.

We use LIBSVM with a radial basis function (RBF) kernel to implement SVM classification [29]. In the four-way pitch accent detection, the one-versus-the-rest mode is adopted to decompose multi-class classification into binary. For NN, we use a three-layer network with a fully connected structure. The sigmoid activation function is chosen for network nodes. The classical backpropagation is adopted to minimize the cross entropy error between outputs and targets. Many options related to the two classifiers, such as the number of hidden nodes, the learning rate, the momentum for NN and the cost, the gamma value of RBF function for SVM, are optimized with the development dataset. The results of the prosodic event detection by the two classifiers are given in Table 4.

We can see from Table 4 that on this task, SVM outperforms NN. Another advantage of SVM is that it has fewer parameters to tune and thus is not as sensitive to them as NN. Therefore, considering efficiency and accuracy, we adopt the SVM as our basic classifier in the following experiments. During the iterative process, the same configuration is used for all SVM classifiers. The posterior probabilities that the auto-context algorithm uses are obtained by mapping the distance between the sample and the classifying hyperplane with a sigmoid function.

6.2 Auto-context algorithm for pitch accent detection

We conduct the following experiments to investigate the ability of the auto-context algorithm to model contextual information for the task of pitch accent detection. The first experiment investigates the effect of prosodic context utilized by auto-context for pitch accent detection. Performance is verified only in independent-syllable conditions, where no acoustic context has been used in the baseline. Based on this, a further investigation is then conducted to identify the effect of different contextual locations, e.g., preceding and following contexts. In the final experiment, acoustic differential features are employed to investigate the combination effect of prosodic context and acoustic context.

6.2.1 *Auto-context algorithm without acoustic context*

To perform this experiment, we select the number of contextual syllables before and after the current syllable to range from $M = 1$ to $M = 5$, where M is the maximum contextual extent. For each selected syllable, the probabilities of all classes were included.

Figure 4a,b shows the average detection accuracy of the auto-context algorithm calculated across the training and test sets of the fivefold cross-validation. For each case, the training set accuracy increases with each iteration, and the wider the contextual range, the better the performance. Correspondingly, the test set accuracy also increases gradually across iterations, as illustrated in Figure 4b. This suggests that the auto-context algorithm is able to iteratively capture informative prosodic context. The larger the

Table 5 Accuracy performance of the auto-context algorithm using contextual information from different locations

Auto-context (%)	None	Preceding	Following	Both
Two-way ($M = 3$)	79.3	80.4	80.1	81.2 ($M = 3$ itr = 3)
Four-way ($M = 2$)	74.2	75.3	75.0	75.7 ($M = 2$ itr = 6)

Here, 'None' means not using auto-context, and 'Preceding', 'Following', and 'Both' refer to the directions of contextual prosodic information.

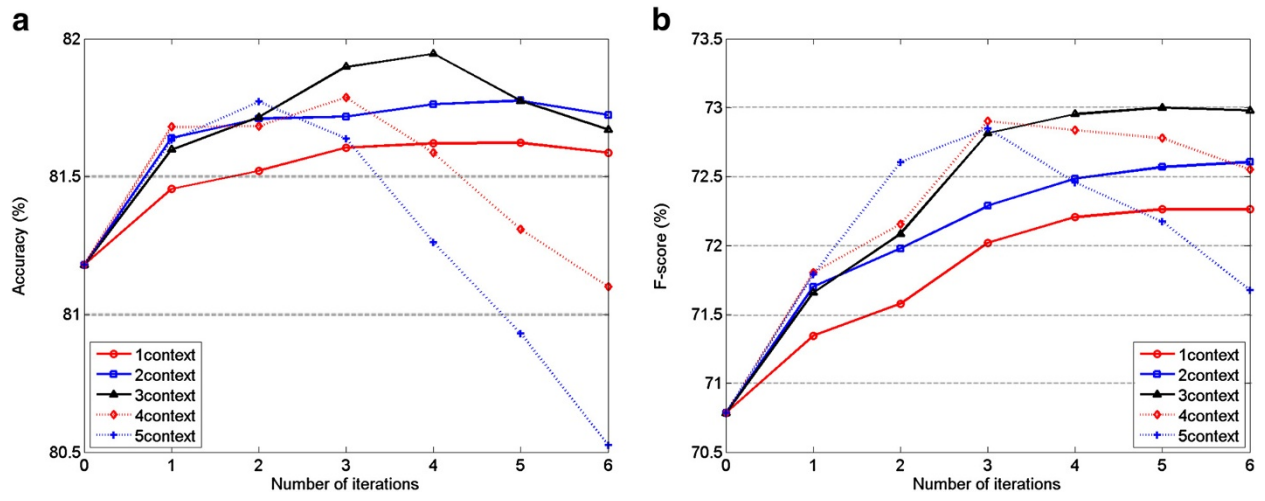

Figure 6 Performance of the auto-context algorithm on test dataset for two-way pitch accent detection with acoustic features. **(a)** Accuracy performance on test data. **(b)** F-score performance on test data.

range is, the greater ability this algorithm has to model these useful relationships.

Additionally, we can see that the greatest performance gain is often obtained in the first iteration. In our experiment, the first iteration produces no less than 1.5% net accuracy improvement (more than 70% of the total improvement) on test data. This is because it is at this stage that the context is first added to a baseline without any prior contextual information. It can also be observed that sometimes there is performance degradation on the test set when the classifiers are iterated too many times, which indicates the over-fitting of the model. To avoid this problem in practice, a development dataset needs to be used to tune and determine an appropriate configuration before each implementation. In our experiment under the fivefold cross-validation scenario, the optimal configuration is determined by the average performance of the development dataset. Final results are obtained under the constraint of the chosen configuration. In order to fully show the change trend, the curves shown in the figures are not preset with any stopping criterion. From Figure 4b, we can also see that $M = 3$ gives the best performance. This is consistent with statistics, showing that most pitch accent events occur once every two or three syllables. The final result is obtained at $M = 3$ and iteration number itr $= 3$, achieving accuracy of 81.2%.

The F-score performance on the training and test datasets are fully shown in Figure 4c,d, respectively. As

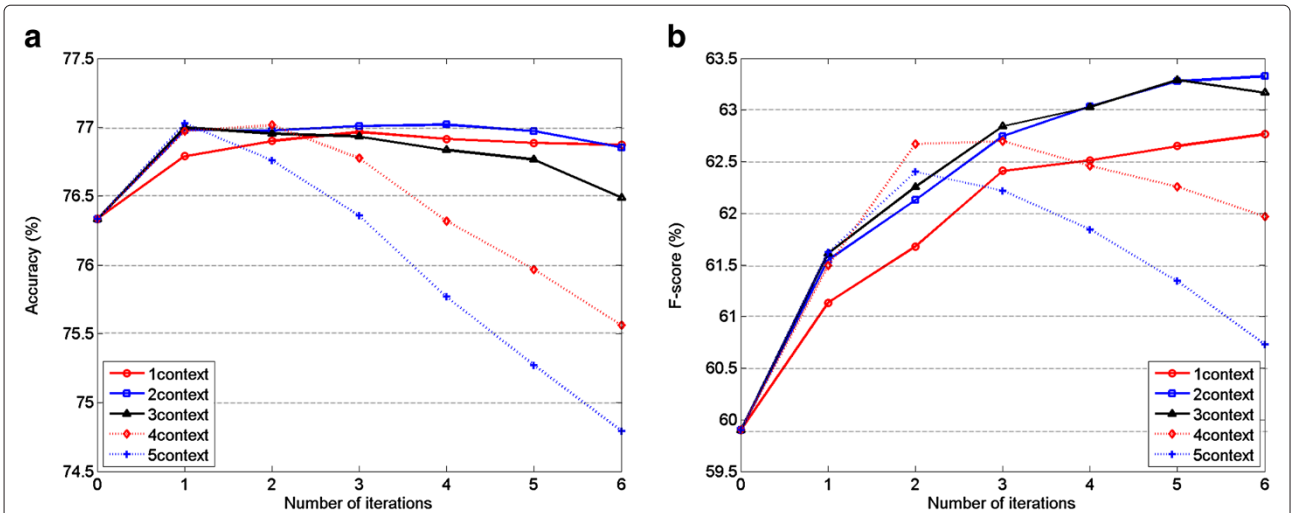

Figure 7 Performance of the auto-context algorithm on test dataset for four-way pitch accent detection with acoustic features. **(a)** Accuracy performance on test data. **(b)** F-score performance on test data.

Table 6 Performance comparison of auto-context, acoustic context, and their combinations

	No context	Acoustic context	Auto-context	Combination
Two-way (%)				
Accuracy	79.3	81.2	81.2 ($M = 3$ itr = 3)	82.0 ($M = 3$, itr = 4)
F-score	66.7	70.8	71.1 ($M = 3$ itr = 4)	73.0 ($M = 3$, itr = 4)
Four-way (%)				
Accuracy	74.2	76.3	75.7 ($M = 2$ itr = 6)	77.0 ($M = 2$, itr = 1)
F-score	54.3	59.9	59.5 ($M = 2$ itr = 6)	63.0 ($M = 24$, itr = 4)

with accuracy, we see a similar increasing trend. The first iteration again gives the most salient improvement, more than 3% net, representing more than 60% of the total improvement. The F-score improvement is more significant than the accuracy improvement. In our experiment, auto-context improves the F-score from 66.7% to 71.1% at $M = 3$ and itr $= 4$ on the test data.

The test accuracy and F-score performance on the four-way pitch accent detection are fully shown in Figure 5a,b. From this figure, we can see that the auto-context can substantially impact the four-way pitch accent detection result as well. In our experiments, it can improve the accuracy from 74.2% to 75.7% at $M = 2$ and itr $= 6$, and improve the F-score from 54.3% to 59.5% at $M = 2$ and itr $= 6$.

6.2.2 *Effect of contextual location*

Auto-context is sensitive to the choice of context. Section 6.2.1 has discussed the performance variation associated with contextual range influence. In this experiment, we conducted further analysis about how different contextual locations impact the auto-context effectiveness. The contextual location was split into preceding and following based on the position relative to the current syllable. We chose an upper contextual limit of $M = 3$ for the two-way task and $M = 2$ for the four-way task, based on the results of the previous experiments. In each experiment, only the preceding or following probabilities are used. The final results are given in Table 5. We see that both the preceding and following contexts are useful for the tasks of two-way and four-way extent detection. As expected, combining both of them gives the best performance.

6.2.3 *Auto-context algorithm with acoustic context*

The auto-context algorithm has the potential to explore relationships between prosodic events across a wider extent and combine these with acoustic information in a unified framework. In this experiment, under the same experiment setups to Section 6.2.1, we implement the auto-context algorithm based on not only local acoustic features but also acoustic context to investigate the combination effect between prosodic context and acoustic context. The testing accuracy and F-score results on the two-way task are shown in Figure 6a,b, and the testing performance of the four-way classification is shown in Figure 7a,b.

The final results are listed in Table 6. From these results, we can see that although the prosodic contexts used by auto-context achieves comparable performance to direct acoustic context when used separately, the combination of them yields further improvement. Based on the feature set that has included acoustic context, auto-context can give a further accuracy improvement on both of the

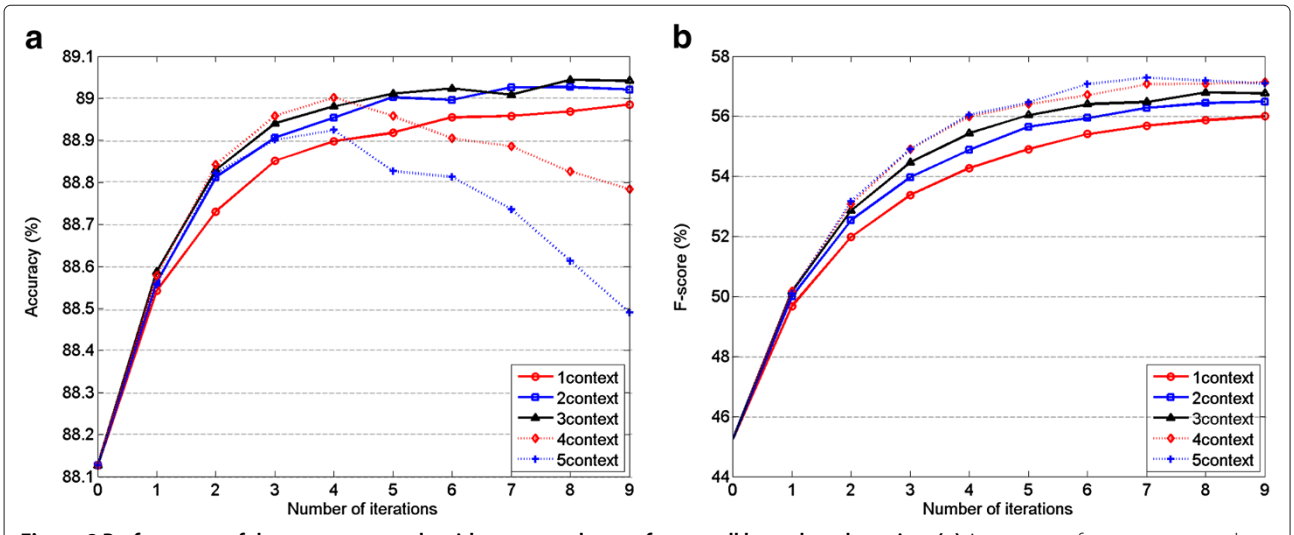

Figure 8 Performance of the auto-context algorithm on test dataset for overall boundary detection. (a) Accuracy performance on test data. **(b)** F-score performance on test data.

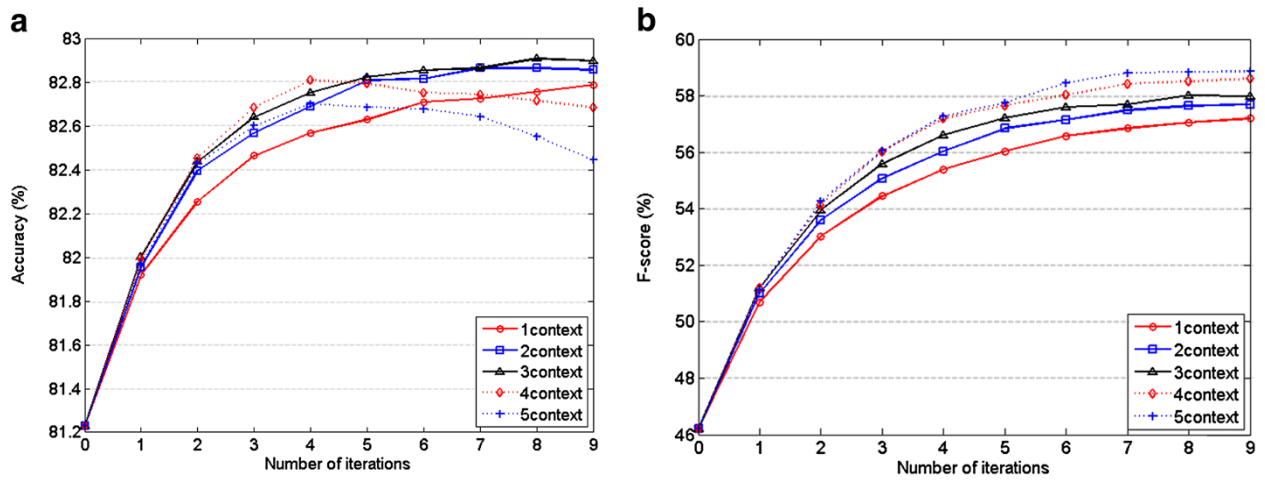

Figure 9 Performance of the auto-context algorithm on test dataset for word-final boundary detection. (a) Accuracy performance on test data. **(b)** F-score performance on test data.

pitch accent detection tasks. It improves the accuracy of two-way detection from 81.2% to 82.0% at $M = 3$ and itr $= 4$, and improves the accuracy of four-way detection from 76.3% to 77.0% at $M = 2$ and itr $= 1$. It also gives a F-score improvement of more than 2% (from 70.8% to 73.0% at $M = 3$ itr $= 4$) for the two-way task and 3% (from 59.9% to 63.0% at $M = 2$ itr $= 4$) for the four-way task. The addition of acoustic context improve the overall performance of auto-context as well, compared to the case of only utilizing prosodic context. For example, without acoustic context information, auto-context can achieve an accuracy of 75.7% on four-way pitch accent detection. When combined with acoustic context, however, the performance improves to 77.0%. These improvements also have been shown in other detection tasks and confirmed by F-score measurements. This demonstrates that prosodic context and acoustic context provide complementary contextual information and that the auto-context algorithm can effectively combine them together.

6.3 Auto-context on boundary detection

This experiment investigates the performance of auto-context algorithm on boundary detection. Using the boundary-related features introduced in Section 5.2, we obtain a syllable-level accuracy of 88.1% and F-score of 45.3% using overall statistics, and a word-level accuracy of 81.2% and F-score of 46.2% using word-final statistics. On this baseline, we implemented the auto-context algorithm across different contextual extents. The syllable-level accuracy and F-score performances are shown in Figure 8a,b, and the corresponding results of the word-level performance are shown in Figure 9a,b.

The final results are listed in Table 7. As we can observe, for boundary detection, the auto-context algorithm yields a nearly 1% improvement using overall statistics and 1.6% improvement using word-final statistics. For the former, it achieves an accuracy of 89.0% at $M = 4$ and itr $= 4$, while for the latter, it achieves accuracy of 82.9% at $M = 3$ and itr $= 7$. Similarly to pitch accent detection, the auto-context algorithm achieved a larger improvement on F-score than the accuracy for the boundary detection task, improving the F-score by about 12% for both statistics. It

Table 7 Performance of boundary detection when using the auto-context algorithm

Auto-context (%)	Accuracy (%)		F-score (%)	
	None	Auto-context	None	Auto-context
SL boundary	88.1	89.0 ($M = 4$, itr $= 4$)	45.3	57.3 ($M = 5$, itr $= 6$)
WL boundary	81.2	82.9 ($M = 3$, itr $= 7$)	46.2	58.9 ($M = 5$, itr $= 7$)

Here, 'None' means not using auto-context.

Table 8 Performance comparisons of n-gram, CRF, and auto-context methods

	N-gram [8]	CRF	Auto-context
Accuracy (%)			
Two-way	80.1	81.1	82.0
Four-way	-	76.9	77.0
SL boundary	89.6	88.2	89.0
WL boundary	84.0	82.2	82.9
F-score (%)			
Two-way	-	72.5	73.0
Four-way	-	62.9	63.0
SL boundary	-	55.1	57.3
WL boundary	-	56.3	58.9

achieves a syllable-level F-score of 57.3% and word-level F-score of 58.9%, respectively.

6.4 Methods comparison

In this section, we compare the performance of our SVM-based auto-context system with two other alternative methods, including an n-gram language model and the CRF approach. For the n-gram approach, we referred to the results of the representative work [8], in which the same two-way pitch accent detection and binary boundary detection are implemented on the BURSC dataset using the syllable-level acoustic features of F0, timing cues, and energy. This work used an NN as the classifier and applied the n-gram language model in the rescoring stage in order to utilize context information. The experiments were conducted without considering acoustic dependencies across syllables, and a five-fold cross-validation is used for the measurement of detection accuracy. By using a 4-gram language model, the accuracy of pitch accent detection was substantially improved from 74.1% to 80.1%. However, there was a slight performance degradation for the boundary detection accuracy from 90.0% to 89.6%. We conducted the CRF-related experiments using the CRF++ toolkit [30] with the same data and acoustic features (include first-order differential features), as were used in the auto-context experiments. The The CRF tool does not support continuous features, so we discretized them with a k-means approach. The linear chain CRF model with bigram mode was used, with the model trained using the limited memory BFGS algorithm. Options like cut-off threshold and number of quantized brackets for acoustic features are also optimally tuned using the development dataset. The final results are listed in Table 8. We can observe that the SVM-based auto-context system achieves better performance than CRF, especially in terms of F-score performance for boundary detection, which surpasses the CRF algorithm by 2%. There is also an advantage in binary pitch accent detection compared with the n-gram language model based on NN, although the performance on binary boundary detection is not improved.

7 Conclusions

In this paper, we introduce a flexible and effective algorithm called auto-context for prosodic event detection. This algorithm uses an iterative approach to incorporate the contextual information to improve prosodic event detection. The probabilities of neighboring syllables are integrated with acoustic features to recursively boost the classification performance of the acoustic models. The experiments on two-way and four-way pitch accent detection and binary boundary detection show that auto-context improves the performance both in terms of accuracy and F-score measurements. The combination of both prosodic and acoustic context together gives the best performance. For two-way pitch accent detection, accuracy is improved from 79.3% to 82.0% and F-score from 66.7% to 73.0%. For four-way pitch accent detection, accuracy is improved from 74.2% to 77.0% and F-score from 54.3% to 63.0%. Similar improvement is also shown for boundary detection. Using the overall statistical method, the detection accuracy is improved from 88.1% to 89.0% and F-score from 45.3% to 57.3%, while using the word-final statistical method, the detection is improved from 46.2% to 58.9%.

Competing interests
The authors declare that they have no competing interests.

Acknowledgements
This work was supported by the National Natural Science Foundation of China under grant nos. 61370034, 61273268, and 61005019.

Author details
[1]State Key Laboratory of Transducer Technology, Institute of Electronics, Chinese Academy of Sciences, No.19 North west Rd., Beijing 100190, China. [2]University of Chinese Academy of Sciences, No. 19A Yuquan Road, Beijing 100190, China. [3]National Laboratory for Information Science and Technology, Department of Electronic Engineering, Tsinghua University, Haidian, Beijing 100084, China. [4]Department of Electrical Engineering, Marquette University, 250 W Wisconsin Ave., Milwaukee, WI 53201, USA.

Received: 14 June 2013 Accepted: 18 December 2013
Published: 28 December 2013

References

1. GA Levow, Context in multi-lingual tone and pitch accent recognition. Paper presented in the 9th International Speech Communication Association conference (Lisbon, Portugal, 4–8 Sept 2005), pp. 1809–1812
2. ML Gregory, Y Altun, Using conditional random fields to predict pitch accents in conversational speech. Paper presented in the 42nd meeting of the Association for Computational Linguistics (Barcelona, Spain, 21–26 July 2004), pp. 677–683
3. Z Tu, X Bai, Auto-context and its application to high-level vision tasks and 3d brain image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **32**(10), 1744–1757 (2009)
4. A Conkie, G Riccardi, RC Rose, Prosody recognition from speech utterances using acoustic and linguistic based models of prosodic events. Paper presented in sixth European conference on speech communication and technology (Budapest, Hungary, 5–9 Sept 1999), pp. 523–526
5. S Ananthakrishnan, SS Narayanan, An automatic prosody recognizer using a coupled multi-stream acoustic model and a syntactic-prosodic language model. ICASSP **1**, 269–272 (2005)
6. X Sun, Pitch accent prediction using ensemble machine learning. Paper presented at the 7th international conference on spoken language processing (Denver, Colorado, 16–20 Sept 2002), pp. 953–956
7. CW Wightman, M Ostendorf, Automatic labeling of prosodic patterns. IEEE Trans. Speech Audio Process. **2**, 469–481 (1994)
8. S Ananthakrishnan, S Narayanan, Automatic prosodic event detection using acoustic, lexical and syntactic evidence. IEEE Trans. Audio Speech Lang. Process. **16**, 216–228 (2008)
9. JH Jeon, Y Liu, Automatic prosodic events detection using syllable-based acoustic and syntactic features. Paper presented at the IEEE international conference on acoustics, speech and signal processing (Taipei, 19–24 April 2009), pp. 4565–4568
10. JH Jeon, Y Liu, Syllable-level prominence detection with acoustic evidence. Paper presented at the 11th annual conference of the International Speech Communication Association (Makuhari, Chiba Japan, 26–30 Sept 2010), pp. 1772–1775
11. GA Levow, Automatic prosodic labeling with conditional random fields and rich acoustic features. Paper presented in ACL-IJCNLP (Hyderabad, 7–12 Jan 2008), pp. 217–223

12. R Fernandez, B Ramabhadran, Discriminative training and unsupervised adaptation for labeling prosodic events with limited training data. Paper presented at the 11th annual conference of the International Speech Communication Association (Makuhari, Chiba, Japan, 26–30 Sept 2010), pp. 1429–1432
13. Y Qian, Z Wu, X Ma, F Soong, Automatic prosody prediction and detection with conditional random field models. Paper presented in the 7th international symposium on Chinese spoken language processing (Tainan, Taiwan, 29 Nov–3 Dec 2010), pp. 135–138
14. K Chen, M Hasegawa-Johnson, A Cohen, An automatic prosody labeling system using ANN-based syntactic-prosodic model and GMM-based acoustic prosodic model. ICASSP **1**, 509–512 (2004)
15. VKR Sridhar, S Bangalore, SS Narayanan, Exploiting acoustic and syntactic features for automatic prosody labeling in a maximum entropy framework. IEEE Trans. Audio Speech Lang. Process. **16**, 797–811 (2008)
16. A Rosenberg, J Hirschberg, Detecting pitch accents at the word, syllable and vowel level. Paper presented in the annual conference of the North American Chapter of the Association for Computational Linguistics: human language technologies (Boulder, Colorado, 31 May–5 June 2009), pp. 81–84
17. M Ostendorf, P Price, S Shattuck-Hufnagel, *The Boston University Radio News Corpus* (Linguistic Data Consortium, Philadelphia, 2013)
18. K Silverman, M Beckman, J Pitrelli, M Ostendorf, C Wightman, P Price, J Pierrehumbert, J Hirschberg, ToBI: A standard scheme for labeling prosody. Paper presented at the international conference on spoken language processing (Banff, Alberta, Canada, 13–16 Oct 1992), pp. 867–869
19. JH Jeon, Y Liu, Automatic prosodic event detection using a novel labeling and selection method in co-training. Speech Commun. **54**(3), 445–458 (2012)
20. K Li, S Zhang, M Li, WK Lo, H Meng, Prominence model for prosodic features in automatic lexical stress and pitch accent detection. Paper presented in the 12th annual conference of the International Speech Communication Association (Florence, Italy, 27–31 Aug 2011), pp. 2009–2012
21. E Zwicker, H Fastl, *Psychoacoustics—Facts and Models* (Springer, New York, 1999)
22. SS Stevens, On the psychophysical law. Psychological **64**(3), 153–181 (1957)
23. S Nooteboom, The prosody of speech: melody and rhythm, in *The Handbook of Phonetic Sciences*, ed. by WJ Hardcastle, J Laver (Wiley, New York, 1997), pp. 640–673
24. P Boersma, A Praat, system for doing phonetics by computer. Glot Int. **5**(9–10), 341–345 (2001)
25. AMC Sluijter, VJ van Heuven, Acoustic correlates of linguistic stress and accent in Dutch and American English. ICSLP **2**, 630–633 (1996)
26. P Taylor, The rise/fall/connection model of intonation. Speech Commun. **15**, 169–186 (1994)
27. P Taylor, The tilt intonation model. Paper presented in ICSLP (Sydney, Australia, 30 Nov–4 Dec 1998), pp. 1383–1386
28. A Rosenberg, Automatic detection and classification of prosodic events. PhD thesis (Columbia University, 2009)
29. RE Fan, PH Chen, CJ Lin, Working set selection using second order information for training support vector machines. J. Mach. Learn. Res. **6**, 1889–1918 (2005)
30. CRF++: Yet another CRF toolkit. http://crfpp.sourceforge.net/. Accessed 25 March 2012

doi:10.1186/1687-4722-2013-30
Cite this article as: Zhao *et al.*: Exploiting contextual information for prosodic event detection using auto-context. *EURASIP Journal on Audio, Speech, and Music Processing* 2013 **2013**:30.