Local Rhyme-based Acoustic Features for Mandarin Tone Recognition

Dinoj Surendran and Gina-Anne Levow

May 25, 2006

Abstract

We investigate the use in Mandarin tone recognition of over two hundred possible local acoustic features based on pitch, overall intensity, and band-passed intensity in the rhyme of a syllable. Features involving pitch height are not as useful as one might expect, showing the need for phrase-level pitch height correction. The intensity contour is useful, particularly when rhyme-initial intensity is subtracted. Intensity in certain medium and high-frequency bands also provides useful information. Unsurprisingly, contour tones are better recognized than level tones using only local features.

In tonal languages, lexical information is carried both by phonemes and by syllable-specific intonation called tones. In the tonal language Mandarin, the five possible tones (high, rising, low, falling, neutral) carry as much information as vowels [1] [2].

Mandarin Tone Recognition is the problem of determining the tone of a syllable. Here, we assume that we know the syllable boundaries.

Acoustic features for Mandarin Tone recognition can be found using duration, pitch, overall intensity, and intensities in various high-frequency bands [3]. However, there are several possible such features. Here we determine a useful subset of them that we can use in further experiments.

For now, we deliberately stick to local features. Other than speaker-normalization, we will not consider features that use information outside the syllable boundary. Furthermore, we will limit ourselves to features computed on the rhyme of a syllable [4] to avoid the effect of syllable-initial consonants.

Pitch and overall intensity measurements were found using Praat [5]. Band energy measurements were found using multi-taper spectral analysis [6] by considering overlapping 20ms chunks of speech every 5ms.

1 Features Considered

The 221 local features we considered were the following.

- duration : Duration of the rhyme in milliseconds.
- # voiced : Number of voiced samples in the rhyme.
- int(F) : mean, gradient, and intercept (all across the rhyme) of the contour energy between F-250 Hz and F$+250$ Hz, for F $=$ 250, 500, $\ldots$, 7500, 7750 Hz. There were $3 \times 31 = 93$ such features.
- We considered three acoustic measures, that we shall refer to as ‘cues’. Each was z-normalized by story before computing any features based on it.

  - pit : logarithm of pitch. Pitch in unvoiced regions was defined using linear interpolation.
  - int : overall intensity
  - int>2k : intensity between 2000 and 8000 Hz.

  Suppose ℓ is the duration of the rhyme in milliseconds. Then, for each cue we had $x_i, i = 1, \dots, \ell$ as the value of the cue i milliseconds into the rhyme. This was used to compute the following $2N + 7$ features, where N=6 was the size of the fixed-length duration-normalized contour used to represent each contour.

  - cue n, where n ranges from 1 to N, is the cue value at the nth point of the duration-normalized contour. The points are equally spaced from the start to the end (inclusive) of the rhyme.
  - D(cue) n, where n ranges from 1 to $N - 1$, is the difference cue $n + 1 -$ cue n. These represent the derivative of the duration-normalized contour.
  - cue mean : mean of $x_1, \dots, x_\ell$ = average cue value across the rhyme. (Note the computation with the original contour, not the duration-normalized contour.)
  - cue med : median of $x_1, \dots, x_\ell$
  - cue max : maximum of $x_1, \dots, x_\ell$
  - cue min : minimum of $x_1, \dots, x_\ell$
  - cue stdv : standard deviation of $x_1, \dots, x_\ell$
  - cue range : cue max $-$ cue min
  - cue grad : Gradient of the line of best fit to $x_1, \dots, x_\ell$.
  - cue icpt : Intercept of said line.

- For each of the above three story z-normalized cues, the following features were computed based on the cue contour minus the cue at the start of the rhyme: contour, mean, median, minimum, maximum, range, standard deviation.
- As above, but with the rhyme-mid value subtracted instead of the rhyme-initial value. and middle

Each feature was z-normalized by story, so all but the durational measures were doubly normalized.

2 Data

We used 20 stories from news broadcasts in the Mandarin Voice of America TDT2 2 corpus [7]. They were automatically segmented, force aligned, and manually checked; see [4] for details. Table 1 shows summary statistics for the four sets involved.

Training was done with 10 stories from a female speaker[1]. Feature selection was done using classification accuracy on a heldout set of 6 stories[2] from a different female speaker. Testing was done with 2 test sets to help separate speaker-dependence issues. Test Set 1 had two stories[3] a male speaker, while Test Set 2 had two stories[4] from the female speaker of the training set. Of course, these stories were not in the training set.

The final efficacy of feature selection was determined using 0-1 classification accuracy on the test sets, primarily Test Set 1.

[1] Training stories: VOM19980630_0730_0002/0043/0091/0136/0191/0216/0248/0268 and VOM19980630_0700_0032/0432.

[2] Heldout stories : VOM19980630_0900_0005/0040/0105/0127/0205/0230.

[3] Test Set 1: VOM19980630_0700_0262/0328.

[4] Test Set 2: VOM19980630_0700_0238/0296.

Table 1: The size and per-class distributions of the four subsets of data used in the experiments.

Set	Fraction					# syllables
	high	rise	low	fall	neut	
Training	21.7	26.5	13.8	32.3	5.6	1275
Heldout	22.0	25.2	14.0	33.6	5.2	943
Test 1	23.6	18.2	12.8	36.8	8.7	242
Test 2	19.3	31.1	12.7	31.3	5.7	212

3 Classification Algorithm

The classification task here was a 5-class problem (labelling each syllable x with its correct tone y) so we created ten 2-class problems in 1-vs-1 fashion for speed [8], each of which used a Support Vector Machine (SVM) [9] with a Radial Basis Function Kernel $K(x, x') = exp(-\gamma||x - x'||^2))$ with Platt scaling [10] to produce pseudo-probabilities.

Each SVM had two parameters: γ and a penalization parameter c. For convenience we used the same γ and c for all ten SVMs; the best such γ, c were found using 3-fold cross-validation on the training set. The estimated probabilities from the ten SVMs were combined to form estimated probabilities across the five classes [11] and the class with highest probability used as the final prediction.

During training, all training examples were weighted in inverse proportion to the empirical (training set) probability of their true class. All experiments were done using LIBSVM [12].

4 Feature Selection

We used a greedy feature selection heuristic to produce a subset S of features. It aims to maximize accuracy using the RBF SVM on a heldout set. It does not guarantee robustness or optimality.

4.1 Bootstrapping Heuristic

Our feature selection procedure requires an initial bootstrapping step. For this we used weights from an ensemble of linear SVMs. As this was a 5-class problem, we created ten 2-class linear SVMs sharing the same penalization parameter c. This time the c was optimized on the heldout set.

Suppose w_{di} is the absolute value of the weight for the d-th feature, $d = 1, \ldots, F$, from the i-th SVM, $i = 1, \ldots, 5(5-1)/2$. If this is high, then the i-th SVM uses the d-th feature a lot.

The mean absolute weight $\bar{w}_d = \frac{1}{10}\sum_{i=1}^{10} w_{di}$ is high if the d-th feature is useful for many SVMs. The maximum absolute weight $w_d^{max} = \max(w_{d1}, \ldots, w_{d10})$ is high if the d-th feature is useful for some SVM.

We ranked all features according to both $\bar{w}$ and w^{max} and then defined the ‘importance’ of each feature to be the average of the two ranks. More ‘important’ features had lower average ranks.

4.2 Algorithm

```
% Initialization

Define ‘importance’ of each feature using approximate method of Section 4.1

% Starting with an empty $S$, add features to it in order of ‘importance’.
%  If heldout-accuracy increases, increment $S$ with the feature.

accu(0) := 0;
$S = \emptyset$
for $i = 1$ to $F$
    $S^*$ := $S \cup \{i$th ‘most important’ feature$\}$
    accu($i$) $=$ RBF SVM accuracy on heldout set using using features in $S^*$
    if accu($i$) $>$ accu($i - 1$)
        $S$ := $S^*$

% Repeat the above step using all features not in $S$

$S_0$ := $S$
$ac_{max}$ $=$ RBF SVM accuracy on heldout set using features in $S_0$
for $j = 1$ to $F - |S_0|$
    $S^*$ := $S \cup \{j$th ‘most important’ feature not in $S_0\}$
    $ac$ $=$ RBF SVM accuracy using features in $S^*$
    if $ac > ac_{max}$
        $ac_{max} = ac$
        $S$ := $S^*$

% In decreasing order of ‘importance’, see if any features in $S$
% can be removed to increase accuracy

$S_1$ := $S$
for $j = 1$ to $|S_1|$
    $S^*$ := $S - \{j$th ‘least important’ feature in $S_1\}$
    $ac$ $=$ RBF SVM accuracy on heldout set using features in $S^*$
    if $ac > ac_{max}$
        $ac_{max} = ac$
        $S$ := $S^*$
```

4.3 Feature Importance

Finally, we recomputed importance, based on our feature subset S. For each feature in S, we defined its importance to be the *de*crease in classification accuracy (on the heldout set) when the feature was removed. If this was negative then the feature should not have been in S in the first place.

For each feature not in S, we defined its importance to be the *in*crease in accuracy when they were

Table 2: Per-tone and overall accuracy using all 221 acoustic rhyme-based features considered here, and the subset `FeatSel` of 21 features chosen by the feature selection heuristic of Section 4.

Accuracy	high	rise	low	fall	neut	overall
New Speaker (Test Set 1)						
all features	35.1	68.2	32.3	74.2	19.1	53.7
`FeatSel`	42.1	79.6	38.7	69.7	19.1	56.6
Training Speaker (Test Set 2)						
all features	70.7	81.8	51.9	84.9	41.7	74.5
`FeatSel`	65.9	75.8	40.1	83.3	41.7	69.8

added. If this was positive, then the feature was useful, and should have been included in S in the first place.

5 Results

When using all features, classification accuracy on 53.7% and 74.5% on Test sets 1 and 2 respectively. (It was 60.3% on the heldout set.) In comparison, [4] obtained accuracy of 68.5% on the same dataset using only local features — however, those were with different folds where training and test sets had syllables from the same speaker.

Tables 3 and 4 show all features ranked with this new measure of importance. It turned out that our chosen subset was first-order optimal, in the sense that no feature not in the subset can be added to it to improve accuracy and no feature in the subset can be removed to improve accuracy.

Table 2 shows the per-class accuracy. The results on the two test sets are very different. The features selected result in higher accuracy on the new speaker (Test Set 1) at the expense of lower accuracy on the old speaker (Test Set 2). More specifically, the changes in accuracy for the first three tones are opposite for the two test speakers.

6 Conclusions

Our two primary observations are that intensity features are more useful than they are generally given credit for, and that future experiments of this type need to be very cautious with their experimental setup as speaker dependence makes a huge difference (even after speaker normalization). More detailed observations can also be made:

- Features involving minimum, maximum-minimum, and intercept, are not useful.
- Intensity above 2kHz is not useful.
- Duration is useful. While the number of voiced samples made no difference, it was not misleading. (Preliminary experiments done for this paper, which had the same speakers shared across training and heldout sets, found it was more useful than duration.)
- Band energy between 6250 and 6750 Hz is the most useful feature. It is particularly useful for rising tone (accuracy on the heldout set drops from 61.3% to 48.3% without it), low tone ($46.2\% \rightarrow 39.4\%$) and falling tone ($77.3\% \rightarrow 72.6\%$). It does not affect recognition rates for the high or neutral tones. We suspect it is really acting as a fricative detector, but have not verified this.

Table 3: ‘Importance’ of the most useful half of 221 rhyme-based features Features in bold were in the final feature subset found in Experiment 1 and appear with the absolute **de**crease in accuracy if they are removed from the subset. Features not in bold appear with the absolute **in**crease in accuracy if they are added to the subset. Accuracy, which is given as a percentage, is on the heldout set when an ensemble of RBF SVMs (with Platt scaling) is used in 1-vs-1 fashion. This table is continued in Table 4.

int(6500)	**5.8**	int 5:5−start	-0.5	int min−start	-1.1
pit stdv	**2.7**	int>2k 6:5−start	-0.5	int range−start	-1.1
int mean−mid	**2.7**	pit med−mid	-0.6	int(5250)	-1.2
int(1000)	**2.4**	int(1500) grad	-0.6	int(250)	-1.2
int 4:5−start	**2.3**	int 5:6	-0.6	int(6750)	-1.2
int 2:5−start	**2.1**	D(int) 5:5	-0.6	D(int) 2:5	-1.2
D(int) 4:5	**1.8**	int 6:5−start	-0.6	int 3:6	-1.2
int max−start	**1.7**	int med−start	-0.6	int min−mid	-1.2
D(int) 1:5	**1.7**	int 1:6	-0.6	int(7750) grad	-1.2
int mean−start	**1.6**	int min	-0.6	int(7750) icpt	-1.2
D(int) 3:5	**1.5**	D(pit) 3:5	-0.7	int(3250) grad	-1.2
pit grad	**1.3**	pit 5:6	-0.7	int>2k 3:6	-1.2
duration	**1.2**	int>2k mean−start	-0.7	int(4250)	-1.3
pit 2:6	**1.2**	int 3:5−start	-0.7	int 6:6	-1.3
int(2000) grad	**1.2**	int>2k range−start	-0.7	int(6750) icpt	-1.3
int(1750)	**0.5**	pit 4:6−mid	-0.8	int stdv	-1.3
pit max−mid	**0.4**	pit stdv−mid	-0.8	int(7500) grad	-1.3
int(2000)	**0.3**	D(pit) 4:5	-0.8	int(7250) grad	-1.3
D(pit) 2:5	**0.1**	int>2k 3:5−start	-0.8	int>2k 4:6	-1.3
pit 3:6−mid	**0.1**	D(pit) 5:5	-1.0	int>2k stdv−mid	-1.3
# voiced	**0.0**	int(3250)	-1.0	pit 6:6	-1.4
int(2500)	-0.1	int(5500)	-1.0	pit icpt	-1.4
int>2k med−start	-0.1	int 4:6−mid	-1.0	int>2k stdv−start	-1.4
pit max	-0.2	int 6:6−mid	-1.0	int range	-1.4
pit 6:6−mid	-0.2	pit 4:6	-1.0	int>2k min−mid	-1.4
pit med	-0.2	int med	-1.0	int>2k 2:6	-1.4
int>2k mean	-0.2	int 1:6−mid	-1.0	int stdv−mid	-1.4
int 4:6	-0.3	int>2k 1:6−mid	-1.0	int range−mid	-1.4
int stdv−start	-0.3	int(1750) grad	-1.1	pit mean−mid	-1.5
int(1500)	-0.4	int(500)	-1.1	pit range	-1.6
pit 5:6−mid	-0.4	pit range−start	-1.1	int(6750) grad	-1.6
pit mean	-0.4	int(4000) grad	-1.1	pit min−start	-1.6
int>2k max−start	-0.4	int 3:6−mid	-1.1	int(1250) grad	-1.6
pit 2:6−mid	-0.5	int(3500) grad	-1.1	D(int>2k) 1:5	-1.6
pit 3:6	-0.5	int(7500) icpt	-1.1	int>2k 2:5−start	-1.6
int(750)	-0.5	int(2250) grad	-1.1	int(2750) grad	-1.6
int 5:6−mid	-0.5	int icpt	-1.1	int>2k 4:6−mid	-1.6

Table 4: This is the second half of Table 3. ‘Importance’ of the less useful half of the 221 rhyme-based features. None appeared in the final feature subset, and thus all of these features appear with the absolute **in**crease in accuracy if they are added to the subset. Accuracy, which is given as a percentage, is on the heldout set when an ensemble of RBF SVMs (with Platt scaling) is used in 1-vs-1 fashion.

int 2:6	-1.6	int(4250) icpt	-2.1	int(5500) icpt	-2.7
int 2:6−mid	-1.6	int(6500) icpt	-2.1	D(int>2k) 5:5	-2.7
int(6250) grad	-1.6	int>2k range−mid	-2.1	int(3750) icpt	-2.8
int>2k 4:5−start	-1.6	int(2500) grad	-2.1	int>2k med	-2.8
int>2k 6:6	-1.6	pit 5:5−start	-2.2	int(5750) icpt	-2.9
int(3000)	-1.7	int(3000) grad	-2.2	int>2k grad	-2.9
pit range−mid	-1.7	D(int>2k) 4:5	-2.2	int(3750)	-3.0
int>2k 1:6	-1.7	int(6000) icpt	-2.2	int(3500) icpt	-3.0
pit med−start	-1.7	int(1250) icpt	-2.2	int(5000) icpt	-3.0
int max	-1.8	D(int>2k) 3:5	-2.2	int(750) icpt	-3.0
pit min−mid	-1.8	int(5500) grad	-2.2	int(3750) grad	-3.1
int(2750) icpt	-1.8	int(6000) grad	-2.2	int max−mid	-3.1
pit min	-1.8	int>2k mean−mid	-2.2	int>2k 2:6−mid	-3.1
int(2000) icpt	-1.8	int(4000) icpt	-2.2	int>2k med−mid	-3.2
int(2750)	-1.9	pit mean−start	-2.2	int(2250) icpt	-3.2
pit 3:5−start	-1.9	int>2k 5:6	-2.2	int(4750)	-3.3
pit stdv−start	-1.9	int(1000) grad	-2.3	int(2500) icpt	-3.3
int grad	-1.9	int(7500)	-2.3	int(5000) grad	-3.3
int(7000) grad	-1.9	int(7250) icpt	-2.3	int(4750) grad	-3.4
int(1500) icpt	-1.9	int(250) grad	-2.3	int(5250) grad	-3.4
int(250) icpt	-1.9	int(6000)	-2.4	int(3000) icpt	-3.4
pit 4:5−start	-1.9	int(6250)	-2.4	int(6250) icpt	-3.4
int(4000)	-1.9	D(pit) 1:5	-2.4	D(int>2k) 2:5	-3.4
int>2k stdv	-2.0	pit 2:5−start	-2.4	int(1250)	-3.4
int med−mid	-2.0	int(7750)	-2.4	int(3250) icpt	-3.4
int(7250)	-2.0	int(5750) grad	-2.4	int>2k icpt	-3.4
int>2k min−start	-2.0	int(4250) grad	-2.4	int(5000)	-3.5
int(500) icpt	-2.0	int>2k 3:6−mid	-2.4	int>2k max−mid	-3.5
pit 1:6−mid	-2.0	int(500) grad	-2.4	int mean	-3.5
int>2k 5:6−mid	-2.0	int>2k range	-2.5	int(4750) icpt	-3.6
int(1750) icpt	-2.1	int(7000)	-2.5	int(5250) icpt	-3.6
pit 6:5−start	-2.1	int(7000) icpt	-2.5	pit 1:6	-3.7
pit max−start	-2.1	int>2k min	-2.5	int(4500) grad	-3.8
int(5750)	-2.1	int>2k 6:6−mid	-2.5	int>2k max	-3.9
int(1000) icpt	-2.1	int(4500)	-2.7	int(4500) icpt	-3.9
int(750) grad	-2.1	int(3500)	-2.7	int>2k 5:5−start	-4.1
int(6500) grad	-2.1	int(2250)	-2.7		

- The band energy between 750-1250 Hz, and 1500-2250 Hz is useful. However, the energy between 1000 and 1500 Hz is one of the most misleading features, and we are unsure what to make of this. More experiments are clearly required.

- The derivative of the intensity contour is useful.

- Features involving overall intensity are more useful when the rhyme-initial intensity is subtracted. It has been suggested[5] that this is capturing the phrase-level intensity contour.

- The standard deviation and gradient of the absolute pitch contour are far more useful than the absolute pitch contour itself. Of the six points in the absolute pitch contour, only the second (one-fifth of the way into the rhyme) makes it into the final subset. Similarly with the derivative of the pitch contour.

- Subtracting the pitch at the start or middle of the rhyme does not help much. However, if one must use a pitch contour, it is better to subtract the mid-rhyme pitch first. Presumably this effect will disappear with appropriate phrase-level pitch correction.

- Median is a useful alternative to mean for pitch-related measurements, which can be quite noisy. This is not required for intensity-based measurements.

6.1 Grouping Related Features

The features chosen by feature selection above may be near-optimal for the training and heldout sets, but, as indicated by the results on Test Set 1, they do not generalize to other speakers very well.

Our 221 features are really divided into several groups of related features, such as ‘intensity contour minus the intensity at the start of the rhyme’ and ‘derivative of the pitch contour’. If several features in a group make it into the final feature subset, it is likely that other features in the group will also be useful. Similarly, if only one feature of a group of several features is in the final subset but does not result in significant accuracy when removed, it can probably be ignored. The following subset of features resulted in accuracy of 57.9% and 71.7% on Test Sets 1 and 2 respectively, both improvements over the corresponding feature-selected values of 56.6% and 69.8% respectively. Table 5 has more details.

- Duration, and Number of voiced samples

- The following parameters computed using the overall intensity contour minus the intensity at the start of the rhyme: contour, derivative of contour, maximum, mean, and mean minus mid-rhyme value.

- The following parameters computed using the pitch contour minus the pitch at the middle of the rhyme: contour, maximum, median, standard deviation, gradient.

- The mean and gradient in bands centered around 1000, 1500, 1750 and 2000 Hz.

- The mean energy in the band centered around 6500 Hz.

- Pitch mean and median.

[5] Thanks to Jennifer Cole for this suggestion.

Table 5: Per-tone and overall accuracy using acoustic features based on the rhyme of a syllable, and two subsets of features. The first subset `FeatSel` is that chosen by feature selection, while the second subset `Related` is that chosen by taking related features into consideration in Section 6.1. For completeness, we also report accuracy on the heldout set.

Accuracy Features	high	rise	low	fall	neut	overall
New Speaker (Test Set 1)						
all features	35.1	68.2	32.3	74.2	19.1	53.7
`FeatSel`	42.1	79.6	38.7	69.7	19.1	56.6
`Related`	45.6	84.1	32.3	70.8	19.0	57.9
Training Speaker (Test Set 2)						
all features	70.7	81.8	51.9	84.9	41.7	74.5
`FeatSel`	65.9	75.8	40.1	83.3	41.7	69.8
`Related`	70.7	74.2	48.1	83.3	50.0	71.7
Heldout Set						
all features	55.1	61.8	40.9	76.3	24.5	60.3
`FeatSel`	68.1	60.5	47.0	77.9	26.5	64.4
`Related`	66.2	64.3	36.4	74.8	20.4	62.0

7 Summary

Out of 221 local acoustic features based on the pitch, duration and intensity (overall and bands) in the rhyme of a syllable, we found a subset of 21 features that, up to changes of one feature, maximized classification accuracy on a heldout set. We then augmented this set to one of 36 features by including related features.

The resulting subset is more robust to speaker change, producing better performance on syllables from a speaker not seen in training or feature selection. While it also results in reduced accuracy (particulary for rising tone) on new syllables from the training speaker, this is probably due to initial overtraining.

With the caveat that this is a small dataset and that speaker dependence is still a huge factor, we can make some tentative conclusions.

The best classified tones with these local features are the rising and falling tones, which are contour tones. Poor recognition of level tones is unsurprising given the lack of contextual normalization. Neutral tones are still not recognized very well using a different speaker, even with band energy measurements.

Energy around 6500 Hz, which probably captures frication, is very useful, as is rhyme duration and the number of voiced samples in the rhyme. When computing features based on the overall intensity in the syllable rhyme, it is better to first subtract the intensity at the start of the rhyme. To a lesser extent, in the absence of other phrase-level pitch normalization, it is better to subtract mid-rhyme pitch from the pitch contour. The mean energy, and the gradient of the band-passed intensity contour, are useful features when the bands have information between 750 and 2000 Hz.

References

[1] Dinoj Surendran and Gina-Anne Levow, “The functional load of vowels in mandarin is as high as that of vowels,” in *Proc. Intl. Conf. Speech Prosody*, 2004, vol. 1, pp. 99–102.

[2] Dinoj Surendran and Partha Niyogi, “Measuring the functional load of phonological contrasts,” Tech. Rep. TR-2003-12, University of Chicago, 2003.

[3] Dinoj Surendran and Gina-Anne Levow, “Band energy features for mandarin tone recognition,” Tech. Rep., University of Chicago, 2006.

[4] Gina-Anne Levow, “Context in multilingual tone and pitch accent recognition,” in *Proceedings of the 9th European Conference of Speech Communication and Technology*, 2005.

[5] Paul Boersma and David Weenink, “Praat: doing phonetics by computer,” *http://www.praat.org*, 2005.

[6] D B Perceval and A T Walden, *Spectral Analysis for Physical Applications*, Cambridge University Press, Cambridge, U.K., 1993.

[7] Charles L Wayne, “Multilingual topic detection and tracking: Successful research enabled by corpora and evaluation,” in *Proceedings of the Language Resources and Evaluation Conference (LREC)*, 2000.

[8] Chih-Wei Hsu and Chih-Jen Lin, “A comparison of methods for multi-class support vector machines,” *IEEE Transactions on Neural Networks*, vol. 13, pp. 415–425, 2002.

[9] Corinna Cortes and Vladimir Vapnik, “Support vector networks,” *Machine Learning*, vol. 20, pp. 273–297, 1995.

[10] John Platt, “Probabilistic outputs for support vector machines and comparison to regularized likelihood methods,” in *Advances in Large Margin Classifiers*, A.J. Smola, P. Bartlett, B. Schoelkopf, and D. Schuurmans, Eds., 2000, pp. 61–74.

[11] Ting-Fan Wu, Chih-Jin Lin, and Ruby C. Weng, “Probability estimates for multi-class classification for pairwise coupling,” *Journal of Machine Learning Research*, vol. 5, pp. 975–1005, 2004.

[12] Chih-Chung Chang and Chih-Jin Lin, “Libsvm : a library for support vector machines,” *Software available at http://www.csie.ntu.edu.tw/∼cjlin/libsvm*, 2001.