Tone Recognition in Mandarin using Focus

Dinoj Surendran, Gina-Anne Levow, Yi Xu

Computer Science Department
University of Chicago, Chicago, IL, USA
dinoj,levow@cs.uchicago.edu,
Phonetics Department
University College London, London, UK
yi@phonetics.ucl.ac.uk

Abstract

Native speakers of Mandarin produce and perceive tones in ways that depend on the focus with which each word is produced [1]. This paper gives one approach to improving tone recognition algorithms using focus, by training different support vector machines on syllables conditional on their position with respect to the focused word in a sentence. In a four-way tone classification task on focus-labelled laboratory Mandarin speech data collected by Xu [2], error rates improve from 15.2% without using focus to 8.7% when using focus. Using the fact that tones on syllables in focused words are especially easy to recognize, we propose a tone recognition algorithm that makes use of focus without requiring focus labels in either training or test set. The algorithm has an error rate of 9.8% on this data set.

1. Introduction

Tones carry much information in Mandarin [3], [4] and speech recognition algorithms would benefit from improved tone recognition algorithms that only use acoustic parameters. Mandarin has four tones ('high', 'rising', 'low', 'falling'), and a neutral tone.

Wang and Seneff [5] considered ways of improving tone recognition using coarticulation, phrase boundaries, and downdrift, but not using focus. We study the complementary problem, of improving tone recognition using focus, as we have access to manually focus-labelled data. In this paper, focus will refer to narrow focus only.

In Xu [2], a large, controlled collection of clean Mandarin speech was elicited from eight native speakers. Each spoke 480 three-word utterances under varying focus conditions. The words were of length 2, 1, and 2 syllables, and the first and fifth syllables always had first tone (high level).

The classification task in this paper is to recognize the tones of the second, third, and fourth syllables of each phrase. There were 11520 such syllables, with equal numbers of the four tones. (There were no syllables with neutral tone.) Owing to test design, syllables were balanced over focus conditions.

We normalized pitch contours of syllables by speaker, syllable duration, and position in syllable. Each syllable was represented by pitch values at $D = 20$ points along its length. Suppose $x_{sj}(d), d = 1, \ldots, D$ is the pitch value at the d-th point of the j-th syllable of the s-th speaker. We used $z(d) = \frac{x_{sj}(d) - \mu_{sd}}{\sigma_{sd}}$ where μ_{sd} and σ_{sd} are the mean and standard deviation of $x_{sj}(d)$ for all j, i.e. all syllables spoken by the same speaker. This normalization did not account for downdrift between the syllables of a phrase, only within syllables.

Thus the entire dataset consisted of 11520 syllables; each syllable represented by a vector in $\mathbb{R}^D$, and had a label from 1 to 4 representing its tone.

All classification results reported here are with four-way cross-validation on the above task. Each fold had $6 \times 480 = 2880$ phrases (8640 syllables) from six speakers in the training set and 960 phrases (2880 syllables) from the remaining two speakers in the test set. No syllables from the same speaker were ever in both test and training sets.

2. Classification Method

All experiments used a support vector machine (SVM) with a linear kernel [6] as this was a fast, robust, state-of-the-art classification method with interpretable results. We expect to have had similar relative results with other classifiers.

We briefly describe how a Linear SVM works on a simple 2-class problem. It is given a set of training examples, where each example is a D-dimensional vector and is labelled as -1 or 1. The Linear SVM algorithm simply determines a function $f(x) = sign(w^T x - b)$, where $w \in \mathbb{R}^D$ and $b \in \mathbb{R}$. The weights w are a linear combination of a subset of the training examples. The vectors in this subset are termed 'support vectors', hence the name.

SVMs can be generalized to n-class, $n > 2$, classification in different ways. In LIBSVM [7], the SVM library we used, such a problem is split into $n(n-1)/2$ binary classification problems whose solutions are combined using a simple voting procedure [8].

The weights w provide valuable interpretable information, as they show how each dimension of the vectors is used in the classification process. For example, Figure 1 shows the weights of the six binary SVMs created in our baseline four-class classification experiment. It shows, for instance, that when distinguishing between syllables with high and rising tone, the most important factor is the pitch near the end of each syllable. Above-average syllable-end pitch is more likely to occur on high-tone syllables than rising-tone syllables.

3. Experiments

3.1. Baseline : Tone Prediction without using Focus

Without using any focus-related information, the error rate when applying a linear SVM to this tone recognition problem is 15.2%. This unsurprisingly low error rate reflects the fact that

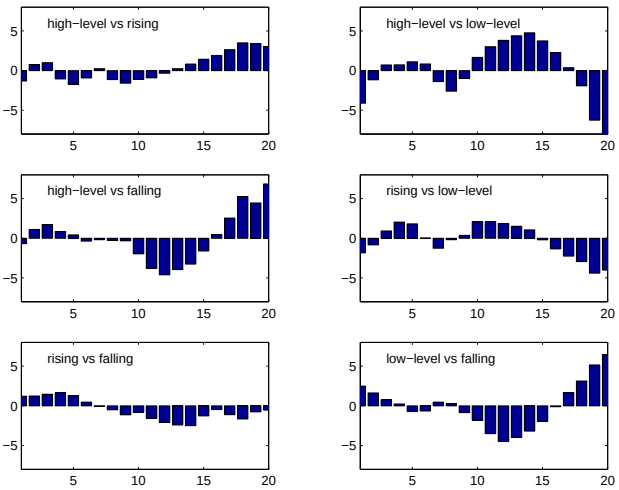

Figure 1: *Weights of the linear SVMs computed for the six binary tone classification problems created while labelling Mandarin syllables with one of four tones. In a bar chart marked "A vs B", a positive weight on the i-th feature/dimension indicate its presence for predicting class A. Negative weights indicate the same for class B.*

this dataset is of clean lab speech. It compares with, for example, baseline error rates of 18.4% for read digit strings reported by Wang and Seneff [5].

3.2. Tone Prediction conditioned on Focus when correct Focus is always known

Our general framework is that the data is actually a collection of sequences, which we will refer to as phrases. In this case, each phrase was a three-word sentence. Syllables and phrases are classified into different focus-dependent classes.

Each phrase can have focus on one word or no words. We will refer to a phrase having neutral focus as a 0-focus phrase, and a phrase with focus on its n-th word as a n-focus phrase. If a n-focus phrase has only n words, it has final focus.

Syllables will be also classified into one of the four classes below, as suggested in [1].

No-focus The syllable is in a phrase with neutral focus.

Pre-focus The syllable is in a word before the focused word of the phrase.

In-focus The syllable is in the focused word of the phrase.

Post-focus The syllable is in a word after the focused word of the phrase.

In our first experiment, we partitioned the set of syllables into four groups, one for each of the focus conditions listed. We trained one SVM per group to recognize tones, and tested it on syllables within the same group. The error rate decreased to 8.7%, a relative error reduction of 42.9% from the baseline.

The error rates on the different subsets of syllables, which are given in Table 1, are very different. Tones of in-focus syllables are recognized with 99.2% accuracy. While we expected syllables in focused words to be easier to recognize, we had not expected them to be this easy, particularly as we had not taken

Table 1: *Classification Error Rates for Four-way Tone Recognition with a Linear SVM in Mandarin for differing conditions of focus. In these experiments, the location of the focused word was known.*

Condition	Error Rate
Combined: Not using focus (baseline)	15.16%
No-focus syllables	7.74%
Pre-focus syllables	7.74%
In-focus syllables	0.80%
Post-focus syllables	18.37%
Combined: Conditional on correct focus	8.66%

coarticulation into account. Clearly, the effect of non-focused syllables on adjacent focused syllables is minimal. Of course, the reverse is not true, and we have high error rates on pre-focus and post-focus words. We expect that accounting for coarticulation would improve those results.

That the error rate for post-focus syllables (18.4%) is much worse than that for pre-focus syllables (7.7%) is indicative of the observation in [9] that articulatory effects are asymmetric; the carryover effect is more than the anticipatory effect. (Another indication of this effect can be found in Figure 1, where weights on pitch values on the latter half of a syllable tend to have higher magnitude.) Furthermore, post-focus syllables have a lower and compressed pitch range. This has two effects that make recognizing their tone difficult. First, the tone on the syllable immediately after focus is on a steep downward ramp that lasts for nearly a syllable, which severely distorts its pitch contour. Second, the compressed pitch range means that, although post-focus syllables are treated separately, there is simply less room for variation in the pitch contours to distinguish between tones.

The error rates for pre-focus and no-focus syllables are identical. While this is a coincidence (their corresponding confusion matrices, shown in Table 2, do not have identical behaviour), it does bring up the theoretical question of whether no-focus and pre-focus syllables behave similarly.

We tested this hypothesis by repeating the above experiment with pre-focus and no-focus syllables grouped in the same class. In other words, a single SVM was created for syllables that were either pre-focus or no-focus. The error rate was still 8.7% (the absolute drop was 0.07%), indicating that the two kinds of syllables behaved similarly.

On the other hand, when we grouped post-focus and no-focus syllables together, the combined error rate increased to 11.7%, indicating that those kinds of syllables did not behave similarly. The error increased further (almost to baseline) to 15.0% when no-focus, pre-focus, and post-focus syllables were all grouped together.

While it has been widely agreed that focus has an impact on the tone of focused syllables, there has been less agreement on whether it affects the tone of non-focused syllables. These results agree with earlier observations of Xu [1] that while pre-focus syllables will have a similar pitch range to no-focus syllables, post-focus syllables will have a lower pitch range than any other kind of syllable.

3.3. Tone Prediction conditioned on Focus when correct Focus is only known during training

The preceding experiments assumed that we knew the correct focus condition of every syllable. As a first step away from that

Table 2: *Confusion matrices corresponding to classification accuracies given in Table 1.* Each matrix's four columns and rows are for high, rising, low, and falling tones, in that order. The (i, j)-th entry shows the number of times a syllable with tone i was classified as having tone j. The top right matrix is a combination of the bottom four matrices.

One SVM for all syllables				SVMs conditioned on focus			
3673	178	167	142	4013	73	15	59
502	1624	112	2	448	1757	25	10
144	38	2631	67	19	22	2789	50
315	4	76	1845	231	17	29	1963
no-focus syllables				pre-focus syllables			
997	29	0	14	743	40	0	17
89	460	9	2	70	716	9	5
0	9	696	15	0	8	454	18
50	2	4	504	29	15	12	744
in-focus syllables				post-focus syllables			
1040	0	0	0	1233	4	15	28
8	551	1	0	281	30	6	3
0	4	716	0	19	1	923	17
9	0	1	550	143	0	12	165

unrealistic assumption, we consider the case where focus is not known on syllables in the test set, but is known on syllables in the training set.

Recall that syllables are part of phrases. We created a new dataset with each data point representing a phrase labeled as n-focus, for $n = 0, 1, 2$ or 3. This is not a particularly general approach, since it was only applicable to this case where all phrases had three words, but it will make our point. Each phrase had a number of features based on its pitch and intensity contours. The error rate of a linear SVM on this 4-way classification task was 28.9%.

Table 3: *Confusion matrix for recognizing the focus condition of a phrase. The (i, j)-th entry has the number of phrases with the i-th focus condition that were recognized as having the j-th focus condition.*

	0-focus	1-focus	2-focus	3-focus
0-focus	605	61	88	206
1-focus	58	871	24	7
2-focus	77	28	759	96
3-focus	295	36	134	495

We computed the focus condition of each syllable based on each predicted focus condition, and then created four linear SVMs based on each condition. The resulting error rate was 9.7%, which is a surprisingly small increase from the corresponding figure of 8.7% when focus is known with 100% accuracy.

One reason for why such a huge error in focus error rate only result in a small effect on tone error rate can be found in Table 3. The phrase focus conditions that were most often confused were neutral and final focus. If a neutral-focus phrase is misclassified as final-focus, nearly all (i.e. all except those in the final word of the phrase) of its no-focus syllables will be treated like pre-focus syllables, and vice versa. Since these kinds of syllables behave similarly, the effect of the misclassification is mitigated.

One problem with this approach, which becomes especially apparent when dealing with phrases with varying numbers of words, is that one has to make a decision as to whether a phrase has neutral focus or not. Since pre-focus syllables behave like neutral-focus syllables, by the argument of the previous paragraph, it is worth folding the neutral-focus and final-focus phrases into one group and hence assuming that all phrases have some focus.

We therefore repeated the above experiment, but with the phrases predicted as having neutral focus given final focus instead. The error rate went *down* to 8.7%. When we repeated the folding using correct focus locations, the error rate went down further, to 8.3%.

3.4. Tone Prediction conditioned on Focus when correct Focus is never known

Unfortunately, the previous set of experiments still required focus values to be known during training. This is usually impossible, so we need a method of determining focus without having any focus-marked phrases to train on.

Having observed that tones were best recognized on syllables with focus, we hypothesized that the confidence of tone prediction could be used to predict which syllables in a phrase were focused. The tone prediction algorithm was the one applied to all syllables regardless of focus condition.

Note that we are referring to the confidence of a prediction, not its accuracy. It is quite possible for confident predictions to be wrong.

With a 2-class SVM, the final prediction function is of the form $f(x) = sign(g(x))$, so the confidence of prediction is $|g(x)|$. With a multi-class SVM, things are not as straightforward. Any SVM can be made to produce not just predictions, but probabilities corresponding to the predictions [10]. For example, it could predict that a syllable has high tone with probability 0.64, rising tone with probability 0.31 and falling tone with probability 0.02 – instead of just predicting that the syllable had high tone.

We took the confidence of a prediction to be the highest probability in the predicted probability distribution, and hypothesized that the word with the most confident syllables was the focused word. Note that we assumed that each phrase now had a focused word.

We trained a linear SVM to recognize tone on all syllables, and then used the confidence of its predictions to predict the location of the focused word of each phrase. (The error rate on recognizing the focus condition of each syllable was 37%.) We created three different linear SVMs conditioned on pre-focus, in-focus, and post-focus syllables based on these predicted focused words, as before, and the resulting error rate was 9.8%. This is still a relative improvement of 35% over the baseline error rate, and does not use labelled focus anywhere in the process. For reference, and comparison to those in Table 2, the resulting confusion matrix is shown in Table 5.

4. Discussion

There are several ways in which one could incorporate focus into a tone recognition algorithm. Here we investigated several possibilities using a focus-marked dataset, and finally proposed an algorithm like this:

1. Partition the training set into n folds
2. For each fold

Table 4: *Confusion matrix for recognizing the focus condition of a phrase, when predicting focus using the confidence of tone prediction. The (i, j)-th entry has the number of phrases with the i-th focus condition that were recognized as having the j-th focus condition.*

	1-focus	2-focus	3-focus
0-focus	131	298	531
1-focus	634	82	244
2-focus	46	736	178
3-focus	101	341	518

Table 5: *Confusion matrix for recognizing tones conditioned on predicted focus, when focus was predicted using the confidence of a non-focus-conditioned, tone recognition classifier. The (i, j)-th entry has the number of syllables with tone i that were recognized as having tone j.*

	high	rising	low	falling
high	3963	69	54	74
rising	431	1728	72	9
low	35	28	2772	45
falling	251	14	47	1928

(a) Train a tone classifier on all syllables in the other $n - 1$ folds of the training set

(b) For each phrase in this fold of the training data set, predict the location of its focused word as follows:

i. Apply the tone classifier to each syllable in the phrase

ii. Using the assumption that the classifier better recognizes syllables in focused words, find the word in the phrase whose syllables whose tones are predicted with the most confidence

iii. Using this predicted focused word, label each syllable in the phrase as pre/in/post-focus.

3. Now each syllable in the training set is marked as pre/in/post-focus

4. Train separate classifiers on the three sets of syllables in the training set.

5. For each phrase in the test set, use the method described above to predict its focused-word and hence label each of its syllables as pre/in/post-focus. Predict the tone of the syllable using the classifier for the corresponding focus condition.

In our experiments we were using cross validation so that the testing set corresponded to a fold of the training set, but we were careful to do it in a manner that did not overlap information from the training and test sets.

Future work will apply the above algorithm to non-laboratory speech, such as news broadcasts. Its efficacy will rely on the assumption that whatever features are used to recognize tones, they are far more effective on focused syllables than any others.

We will also investigate using the output of the above algorithm as the bootstrapping step of a EM-type algorithm that simultaneously recognizes tones and focus.

5. Conclusions

We have presented the results of several experiments on a focus-marked corpus that offer insight on how to incorporate the use of focus in tone recognition. Our final algorithm used these insights to reduce the error rate of tone recognition from 15.2% to 9.8% without using any marked focus information. We also presented further evidence in favor of some theoretical linguistic questions, such as focus affecting post-focused syllables, and the asymmetry of coarticulation.

Additional information can be found at the project website http://people.cs.uchicago.edu/∼dinoj/projects/tonefocus

6. Acknowledgements

We would like to thank Chih-Jen Lin and Chih-Chung Chang for writing LIBSVM, and to Tzu-Kuo Huang for a utility to determine the primal weights for the resulting SVMs. Thanks also go to Randy Landsberg and Mark SubbaRao for access to computing facilities. Funding came from NSF Grant 0414919.

7. References

[1] Xu, Y., Xu, C. X., Sun. X. “On the temporal domain of focus”, Proc. Intl. Conf. Speech Prosody, Nara, Japan. 1:81–94, 2004.

[2] Xu, Y. “Effects of tone and focus on the formation and alignment of f0 contours”, J. Phonetics 27:55–105, 1999.

[3] Surendran, D. and Levow, G. “The functional load of vowels in Mandarin is as high as that of vowels”, Proc. Intl. Conf. Speech Prosody 1:99–102, 2004.

[4] Surendran, D. and Niyogi, P. “Measuring the functional load of phonological contrasts”, Tech. Report TR-2003-12, Univ. of Chicago Comp. Sci. Dept., 2003.

[5] Wang, C. and Seneff, S. “Improved tone recognition by normalizing for coarticulation and intonation effects”, Proc. Intl. Conf. Sp. Lang. Proc. (ICSLP), 83–86, 2000.

[6] Cortes, C., Vapnik, V. “Support vector networks”, Mach. Learn. 20:273–297, 1995.

[7] Chang, C-C. and Lin, C-J. “LIBSVM : a library for support vector machines”, 2001. Software available at http://www.csie.ntu.edu.tw/∼cjlin/libsvm

[8] Krebel, U. “Pairwise classification for support vector machines”, in B. Scholkopf, C. J. C. Burges, and A. J. Smola, editors, Advances in Kernel Methods — Support Vector Learning, p255-268, MIT Press, Cambridge, MA, 1999.

[9] Xu, Y. “Contextual tonal variations in Mandarin”, J. Phonetics 25:62–83, 1997.

[10] Wu, T.F., Lin, C-J., Weng, R.C., “Probability estimates for multi-class classification for pairwise coupling”, J. Mach. Learn. Res. 5:975–1005, 2004.