Learning Score Structure from Spoken Language for A Tennis Game

Qiang Huang and Stephen Cox

School of Computing Sciences
University of East Anglia, Norwich, NR4 7TJ, UK
h.qiang@uea.ac.uk, s.j.cox@uea.ac.uk

Abstract

We describe a novel approach to inferring the scoring rules of a tennis game by analysing the chair umpire's speech. In a tennis match, the chair umpire, amongst other tasks, announces the scores. Hence his or her speech is the key resource for inferring the scoring rules of tennis, a task that can be accomplished by correlating the events on the court with these score announcements. In this work, the learning procedure consists of two steps: speech recognition followed by rule inference. For speech recognition, we use a two coupled language models one for words and one for scores. The first makes make use of the internal structure of a score, the second, the dependency of a score on the previous score. For rule inference, we utilize a multigram model to segment the recognised score streams into variable-length score sequences, each of them corresponding to a game in a tennis match. The approach is applied to four complete tennis matches, and shows both enhanced recognition performance, and a promising approach to inferring the scoring rules of the game.

Speech recognition, rule inference, multigram model

1. Introduction

Automatic information acquisition and knowledge inference is essential for the development of a machine that aims to interact intelligently with humans. Our long-term goal is to build such a system. However, our initial work is concerned with analysing tennis games, where there are a small number of well-defined "events" and the rules of interaction are simple. For present purposes, we use only the audio recording (the soundtrack) of the game—later work will combine this information with video information. Audio information in a tennis game consists of a number of audio events, such as the sound of the ball being hit, the line judges' shouts, crowd noise, the chair umpire's speech, commentators' speech etc. Identification of all such events is useful information for analysing the game [3, 4], but the score announcements by the umpire are essential for inferring the scoring rules.

Recent work [8, 2, 5, 9, 7] has made use of audio and visual information to analyse sports games. However, this work has focused mainly on topics such as scene segmentation [5], event classification[8, 9], and identifying significant events [2, 7] using low-level features. In this paper, we attempt to infer higher-level information by analysing and processing the speech signal from the umpire. The chair umpire, amongst other tasks, announces the match scores, states whether serves are "in" or "let", announces challenges etc., and so his or her speech is the key resource for inferring the game rules. It is quite likely that many people who understand the scoring system of tennis have done so by correlating the umpire's score announcements with the events on the court, rather than by learning it explicitly from a teacher. This motivates us to attempt to automate a machine to do it.

However, recognition of the scores is not easy because of the poor quality of the umpire's speech on the soundtrack:

- crowd noise often obscures the speech, partially or completely;
- commentators' voices often overlap with the speech.
- the duration of umpire's speech is quite short, usually less than 1s.

To tackle these problems, we set constraints on the design of the speech recogniser and we use a pair of coupled language models. One model makes use of the dependency of words within scores, the second the dependency of a score on the previous score, and the models are allowed to influence each other. Finally, for score rule inference, we employ a multigram model [1] to divide a long sequence of recognised scores into game segments, from which we can extract the scoring scheme. Figure 1 shows a block diagram of our approach. The details will be presented in the next sections.

2. Theoretical Framework

Given a sequence of acoustic observations from the umpire, O, our approach is to find the most likely sequence of recognised words W^*, and the associated most likely score sequence S^*:

$$(S^*, W^*) = \arg\max_{S,W} \Pr(S, W|O). \qquad (1)$$

We can approximate the righthand side of equation 1 by:

$$\begin{aligned} \Pr(S, W|O) &\propto \Pr(O|S, W) * \Pr(S, W) &(2)\\ &\propto \Pr(O|W) * \Pr(W) * \Pr(S|W) &(3)\end{aligned}$$

where the first two terms, $\Pr(O|W)$ and $\Pr(W)$, respectively represent the acoustic model and the language model, and $\Pr(S|W)$ is the probability of a score sequence hypothesis S given a word sequence hypothesis W.

Let S_i be a particular sequence of N_i scores $\{S_i, i = 1, \cdots, N_i\}$. We can write

$$\Pr(S|W) \propto \Pr(W|\{S_i\}) \Pr(\{S_i\}) \qquad (4)$$

S^* and W^* can be obtained by maximising jointly the likelihood of the word sequence W and the set $\{S_i\}$:

$$S^* = S_i^* = \arg\max_{\{S_i\}} \Pr(W|\{S_i\}) \Pr(\{S_i\}). \qquad (5)$$

The likelihood of the data, $\Pr(W|\{S_i\})$, measures how well the data fits a given set $\{S_i\}$. The second term in equation 5 evaluates the likelihood of the set $\{S_i\}$ itself [1].

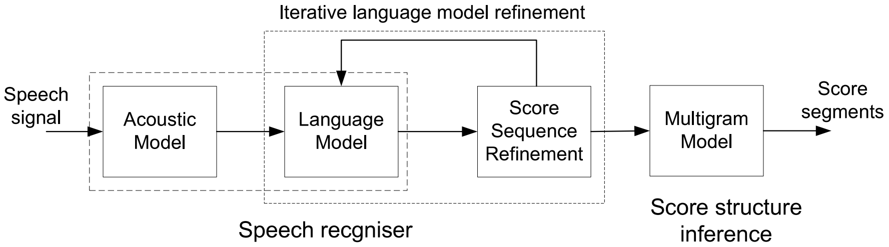

Figure 1: Overview of the system used in this paper. It consists of two main parts: speech recognition and score structure inference

2.1. Design of Speech Recogniser and Language Models

As commented earlier, recognition of the umpire's speech is difficult because of factors such as interference from crowd noise and commentator's speech, as well as the diversity of voices and accents encountered. Our acoustic models are standard monophone Gaussian mixture models trained using a small number of manually labelled umpire's speech segments. We obtain our recognition performance improvement by using two coupled language models, one for word sequences and one for score sequences, that are both iteratively learnt, as indicated in Figure 1. For decoding, we use a word bigram model of every possible set of two vocabulary words. This model is built from the recogniser output obtained from a decoding using a simple word loop i.e. we do not assume "a priori" the syntax of the words within a score. However, this output has many errors because of the lack of constraints on the speech recogniser. We therefore also use a bigram "score language model" which estimates the probability $\mathrm{Pr}\ (\mathrm{score}_i|\mathrm{score}_{i-1})$. This model is also built from recogniser output. This provides considerable constraints on score sequences: for instance, if *score*$_i$ is *fifteen-all*, *score*$_{i+1}$ can only be *thirty-fifteen* or *fifteen-thirty* (assuming, of course, that our current score language model is accurate). After each recognition pass, we re-score the N-best hypotheses (we used $N = 10$) from the word model using the score-language model. The re-ordered hypotheses are then used to re-estimate the word bigram model, which is then used again for recognition, and also to re-estimate the score language model. This is depicted graphically in Figure 2.

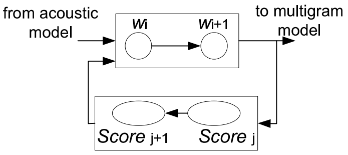

Figure 2: Coupled word and score language models.

When there is no significant change in recognition performance, iteration is stopped, and the recognised score sequences are stored.

2.2. Score Structure Inference

The syntax of the scores in a tennis game is limited and well-defined, so that the same sequences of scores tend to occur in many games. In this section, we describe the use of a multigram model to perform segmentation by identifying commonly occurring score-sequences, which are assumed to represent games.

Let $W = w(1) \cdots w(t) \cdots w(T)$ denote a stream of T scores, and S denote a possible segmentation of W into q sequences of scores $w(1) \cdots w(q)$. The likelihood of the stream of scores W associated with a certain segmentation S is computed as:

$$L(W,S) = \prod_{t=1}^{t=q} Pr(w(t)) \tag{6}$$

Our aim is to find the most likely segmentation of E

$$L^*(W) = \max_{S \in \{S_i\}} L(W,S) \tag{7}$$

where $\{S_i\}$ is the set of all possible segmentations of W into sequences of scores. The multi-gram model is hence fully defined by a set of parameters Θ consisting of the probability of each score sequence $S_i \in \boldsymbol{V}$, where $\boldsymbol{V} = \{S_1, \cdots, S_N\}$, a dictionary containing all the sequences of scores. To compute the set of parameters Θ from a training corpus W, an iterative Maximum Likelihood (ML) is used through an Expectation Maximization (EM) algorithm. For details, the reader is referred to [1].

Estimation of the model parameters is done using an iterative forward-backward procedure [1]. It relies on the estimation of a forward variable α and a backward variable β, which are defined as the likelihood of the partial observed stream of events $W_{(1)}^{(t)}$ and $W_{(t+1)}^{(T)}$, respectively. Figure 3 shows how $\alpha(t)$ and $\beta(t)$ are estimated (Training), followed by segmentation (Decoding) using the Viterbi algorithm [6].

3. Data and Experimental Set-up

We used sound tracks from four complete tennis matches, three men's singles (MS) and one mens' double(MD) from Wimbledon Open in 2008. Table 1 lists the details of these data.

Table 1: Experimental data from four tennis matches

	MS(1)	MS(2)	MS(3)	MD
Duration of the match (Min.)	320	290	120	140
Total length of umpire's speech (s)	401	403	171	176
# umpire's speech fragment	389	387	163	164

From the four matches, we obtained 1103 umpire's speech fragments. As this paper is concerned with score structure inference, we do not consider here the problem of automatic extraction of the umpire's speech from the soundtrack, and we use manually labelled data. Ten examples of each of the eight vocabulary words are selected from MS(1) to train a monophone based acoustic model using HTK. The other 1023 speech fragments are used for testing.

Training

1. **Recursion formula for the variable α**

   for 1⩽t⩽T:
   $\alpha(t) = \sum_{l=1}^{n} \alpha(t-l)p([w(t-l+1)\cdots w(t)])$,
   with $\alpha(0) = 1$ and $\alpha(t) = 0$ for $t < 0$.
   (n is the maximal length of a segment.)

2. **Recursion formula for the variable β**

   for $1 \leq t < T$:
   $\beta(t) = \sum_{l=1}^{n} p([w(t+1)\cdots w(t+l)])\beta(t+l)$,
   with $\beta(T) = 1$ and $\beta(t) = 0$ for $t > T$.

3. **Parameter re-estimation**

   for a sequence S_i of l events,
   $\theta_i^{(k+1)} = \frac{\sum_{t=1}^{T}\sum_{l=1}^{n}\delta(t,l,i)\alpha^{(k)}(t-l)\beta^{(k)}(t)}{\sum_{t=1}^{T}\alpha^{(k)}(t)\beta^{(k)}(t)}$

   where

$$\delta(t,l,i) = \begin{cases} 1 & if\ [\mathrm{w}_{(t-l+1)}\cdots w_{(t)}] = S_i \\ 0 & otherwise \end{cases}$$

Go back to Step 1 for *N* iterations.

Decoding

1. **Initialization**
   $\delta_1(i) = p([w(1)\cdots w(1+i-1)])$
   $\psi_1(i) = 0 \quad 1 \leq i \leq n$
2. **Recursive**
   $\delta_t(j) = \max_{1\leq i\leq n}[\delta_{t-1}(i)]p([w(t)\cdots w(t+j-1)])$
   $\psi_t(j) = arg\max_{1\leq i\leq n}[\delta_{t-1}(i)]$
   $(2 \leq t \leq T, 1 \leq j \leq n)$
3. **Traceback** (refer to [6])

Figure 3: Estimation of parameters for multigram model

As we mentioned in Section 1, our goal is to infer the score structure of a tennis game in a way that mirrors the process of a human learning how sentences are organized after acquiring a few words. We hence assume that the eight words used in tennis scoring, {*love*, *fifteen*, *thirty*, *forty*, *deuce*, *advantage*, *all*, *game*} are known *a priori*. After the recognition process described in section 2.1, the final complete sequence of scores is processed by the multigram model, which finds the most likely segmentation of this sequence into sequences which are assumed to represent games. In practice, the umpire makes other announcements (such as announcing when play will begin, who is to serve etc.) that are not to do with scoring, and these generally contain more than two words. Hence, we remove recognition output containing more than two words prior to using the multigram model.

Performance is evaluated in two ways: speech recognition accuracy, and the effectiveness of the score sequence segmentation. A single error in an early score means that many subsequent scores may be incorrect, and hence even with high word accuracy, the performance of the multigram technique in finding the correct score sequence is rather low. As our goal at this stage is to infer the overall structure of the match rather than score it precisely, we use a metric that focuses on the quality of the segmentation of the stream into games. We note the identity of the first score in each game segmentation produced by the multigram model. The only “legal” scores possible at this point in the game are the two initial scores *fifteen-love* or *love-fifteen*. Hence we can measure the quality of the segmentation using the following definitions of precision (P), recall (R), and F-score (F):

$$P = \frac{\text{\# initial scores detected}}{\text{\# Game starts detected}} \tag{8}$$

$$R = \frac{\text{\# initial scores detected}}{\text{Actual \# game starts}} \tag{9}$$

$$F = \frac{2PR}{P+R}. \tag{10}$$

Table 2: Score recognition accuracy (%)

(%)	**Word loop**	**Iter. 1**	**Iter. 2**	**Iter. 3**	**Iter. 4**	**Iter. 5**
Acc. rate	43.30	47.70	48.97	49.85	50.24	50.34
Imp. rate	-	+10.2	+13.1	+15.1	+16	+16.3

Table 3: Confusion matrix of recognised words after iterations. Columns are actual score words, rows are recognised score words

	advan.	all	deuce	fif.	fort.	game	love	thir.
advan.	41	0	0	5	15	0	0	3
all	0	89	0	0	13	1	30	0
deuce	0	5	78	0	0	5	0	0
fif.	0	0	0	325	21	0	0	53
fort.	11	1	0	29	149	0	0	19
game	23	0	0	0	0	99	23	0
love	0	44	0	0	1	12	213	0
thir.	7	0	0	17	26	0	0	268

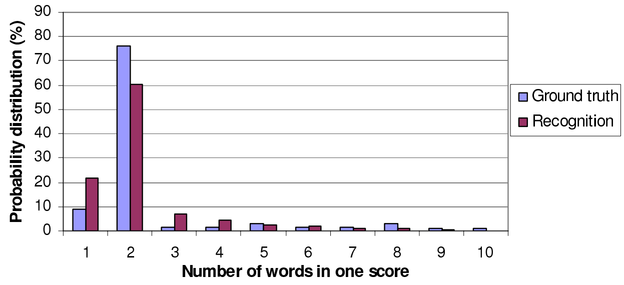

Figure 4: Probability distribution of the number of words in a single score announcement

4. Result Analysis

Table 2 shows the recognition performance as the language models are refined during iteration. After five iterations, a relative performance improvement of 16.3% is achieved. Because our vocabulary consists of only the eight words used in scoring, many words in some longer umpire announcements (e.g. announcements about the state of the match, or about challenges etc.) cannot be correctly recognised, so that overall recognition performance is low. But, for the recognition of scores (shorter speech fragments), the recognition performance is good (77.33% word accuracy). Table 3 shows the confusion matrix of the score words. The probability distribution of the number of the words in the actual scores and in the recognised scores

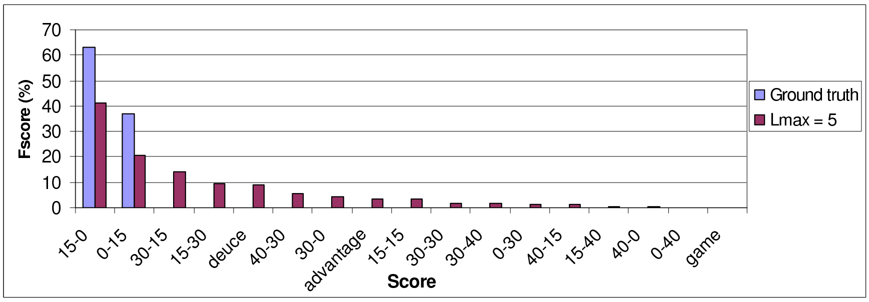

Figure 6: F-score distribution of the first-place score of the ground truth and that of using multigram model with $L_{max} = 5$ on the recognised scores

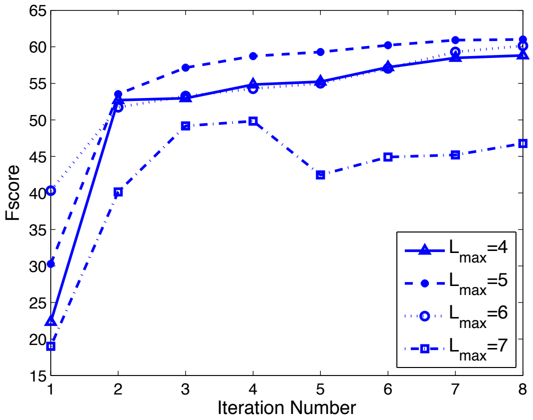

Figure 5: Probability distribution of the first-place score after using multigram model with different L_{max} on the segments

is shown in figure 4. It can be seen that the majority of scores (more than 80%) contain only one or two words.

When using the multigram model, the maximum length L_{max} of a sequence of scores found by the multigram model was set in the range four to seven. Because a segmented sequence represents the sequence of scores in a game, each game can contain at most L_{max} scores. Figure 5 shows the F-score after using the multigram model to segment the recognised score sequence with different values of L_{max}. This shows that $L_{max} = 5$ gives the best performance in game segmentations (as measured by first-place scores). In fact, the mean number of scores in a game (from the ground truth data) is 5.45 (after removing the longer umpire's announcements), which is close to the value of L_{max} we used to obtain the best performance. Figure 5 also shows that the F-score improves with increasing iterations, apart from $L_{max} = 7$: this could be caused by poor parameter estimation due to data sparsity, since the number of possible 7-grams of scores is large.

Figure 6 shows the values of the F-score for the first score detected in a game for $L_{max} = 5$, compared with the ground truth. There are 17 different terms listed on the x-axis. The three terms, "deuce", "advantage", and "game", represent all recognised scores that contain one of these three words. The distribution of first scores is dominated by the two initial scores *fifteen-love* and *love-fifteen*, which is correct: other scores are errors.

5. Conclusion and Future Work

We have developed a promising novel framework for inferring the scoring system in a tennis match using information from the umpire's speech on the soundtrack. Although there are many recognition errors caused by overlapping and interfering speech and noise, we obtain robust recognition performance by the iterative use of two coupled language models. By applying a multigram technique to our recognition output, we obtain reasonable segmentation results. Although we are not yet at the stage where we can infer the complete score syntax accurately, the results are promising. In our future work, we will add information from the video stream, which should increase segmentation performance and hence enable us to infer the score structure more accurately.

6. References

[1] S. Deligne and F. Bimbot, "Inference of variable-length linguistic and acoustic units by multigrams," *Speech Communication*, vol. 23, pp. 223–241, 1997.

[2] Y. Gong, M. Han, W. Hua, and W. Xu, "Maximum entropy model-based base baseball highlight detection and classification," *Computer Vision and Image Understanding*, vol. 96, pp. 181–199, 2004.

[3] Q. Huang and S. Cox, "Hierarchical language modeling for audio events detection in a sports game," in *Proceedings of ICASSP'10*, March 2010, pp. 2286–2289.

[4] ——, "Using high-level information to detect key audio events in a tennis game," in *Proceedings of InterSpeech'10*, 2010, pp. 1409–1412.

[5] S. Lefevre, B. Maillard, and N. Vincent, "3 classes segmentation for analysis of football audio sequences," in *IEEE Int. Conf. on Digital Signal Processing*, 2002, pp. 975–978.

[6] L. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," in *Proceedings of the IEEE*, 1989, pp. 257–286.

[7] Y. Rui, A. Gupta, and A. Acero, "Automatically extracting hightlights for tv baseball programs," in *Proceedings of Int. Conf. Multimedia*, 2000, pp. 105–115.

[8] M. Tien, Y. Wang, and C. Chou, "Event detection in tennis matches based on video data mining," in *IEEE Int. Conf. on Multimedia and Expo*, 2008, pp. 1477–1480.

[9] J. Wang, C. Xu, and E. Chong, "Automatic sports video genre classification using pseudo-2d-hmm," in *Proceedings of the 18th International Conference on Pattern Recognition*, 2006, pp. 778–781.