Understanding How Non-Experts Collect and Annotate Activity Data

Michael D. Jones
Naomi Johnson
Kevin Seppi
Lawrence Thatcher
Brigham Young University
Provo, Utah, 84602, USA
jones@cs.byu.edu,snjohnson789@gmail.com,k@byu.edu,
lwthatcher@msn.com

UbiComp/ISWC'18 Adjunct,, October 8–12, 2018, Singapore, Singapore
ACM 978-1-4503-5966-5/18/10 ... $15.00.
https://doi.org/10.1145/3267305.3267507

Abstract
Training classifiers for human activity recognition systems often relies on large corpora of annotated sensor data. Crowd sourcing is one way to collect and annotate large amounts of sensor data. Crowd sourcing often depends on unskilled workers to collect and annotate the data. In this paper we explore machine learning of classifiers based on human activity data collected and annotated by non-experts. We consider the entire process starting from data collection through annotation including machine learning and ending with the final application implementation. We focus on three issues 1) can non-expert annotators overcome the technical challenges of data acquisition and annotation, 2) can they annotate reliably, and 3) to what extent might we expect their annotations to yield accurate and generalizable event classifiers. Our results suggest that non-expert users can collect video and data as well as produce annotations which are suitable for machine learning.

Author Keywords
data labeling, efficient data collection

ACM Classification Keywords
H.5.m [Information interfaces and presentation (e.g., HCI)]: Miscellaneous

Introduction

Many systems for recognizing human activities in data streams have appeared in the recent literature (see [7, 1] for two recent examples). The construction of these systems follows a pattern consisting of: collect data, annotate data, and use a machine learning algorithm to learn an event classifier. The process depends on annotated sensor data used an input to the machine learning algorithm.

Classifier performance improves as more annotated data are used in training. Improved performance might mean more accurate classification of specific events personalized to a specific individual based on only that individual's data. Or it might mean better generalization over a group of users based on data collected from many people.

In either case, collecting and annotating data is a tedious time-consuming task. Crowd sourcing potentially increases data collection and annotation capacity. Lasecki et al. [11, 12] have constructed systems for crowd sourced annotation of activity data. However more work is needed to understand the entire process of data collection and annotation by non-expert crowd workers.

For our purposes a "non-expert" user is a person who has no prior experience collecting data, annotating events in data and generating classifiers from annotated data sets.

In this paper we present preliminary findings of our ongoing investigation of data collection and annotation based on the experiences of 16 participants in 4 studies. Our purpose is to understand how non-experts collect and annotate data as well to understand the classifiers learned from the collected data and annotations.

In our study of event annotation by non-experts we will use video synchronized with data. Pairing video with data provides the novice annotator with a visual reference for quickly identifying events in the data stream. For example, people can easily recognize a person walking in a video clip but, without experience or training, may have a harder time recognizing a person walking from raw data in an accelerometer trace. Tools like ANVIL [9] ChronoViz [6] and ELAN [16] provide this kind of annotation interface. While additional work is needed to collect video with data and the roles of video and data in annotation tasks has not been carefully investigated, in this paper we assume that both video and data are important for non-expert users.

Our results suggest that non-experts can gather data and video–and can generate accurate, consistent annotations. We demonstrate that event classifiers generated from these annotations can be effective and general.

A Process for Building an Event Classifier

Before describing studies in which non-experts collect and annotate data for event classifiers, we describe the entire process of data collection and annotation as envisioned for this paper. Our purpose is not to propose a novel process for creating event classifiers, but instead to understand how non-experts collect and annotate data for this process.

We used a custom data-logger device to collect the accelerometer and gyroscope sensor data used in this paper. This data, with the assistance of a synchronized video stream, are then annotated using a "Video Annotation Tool" (VAT) that we made for this purpose. The data-logger and VAT give us a data collection and annotation pipeline which we can use in studies involving novice users. While we used our own annotation tool, VAT, other tools like ANVIL, ELAN or ChronoViz could likely have been used with similar results.

The data-logger uses an Invensense MPU-9250 to record

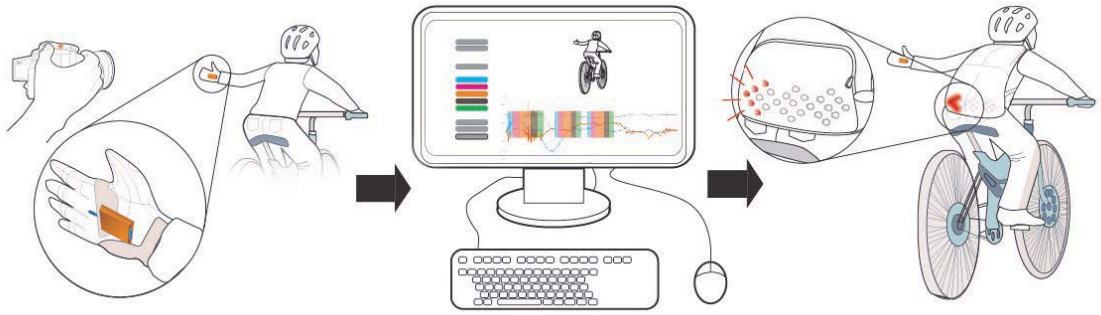

Figure 1: The process for collecting data using the data-logger and then labeling the events in VAT to build an event classifier for bicycle hand signals. We seek to understand the experience of using this kind of system from a non-expert perspective.

acceleration and rotation in 3 axes each. We sample acceleration and rotation each at 25.6 samples per second. The data-logger is similar to motion sensors found in many smartwatches and smartphones. Video was collected at 29.92 frames per second. The VAT tool is a Javascript application which runs in most web browsers.

Figure 1 illustrates the labeling process in the context of a system for controlling LED turn signal for cyclists. The system converts bicycle hand signals into LED signals on the cyclist's backpack. This example will be used to explain the data collection and labeling process in the subsections which follow.

Figure 2: The data-logger attached to a bike glove. The data-logger is secured in the orange housing.

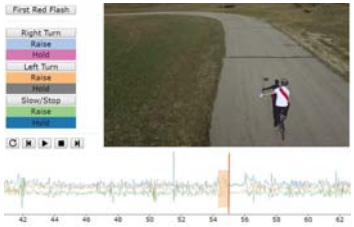

Figure 3: The interface for labeling events in data using a synchronized video feed. The video frame is synchronized with the red vertical line in the data stream.The sequence of data marked in orange is an event region marked as a “Raise” event

Record and Synchronize Video and Data
To collect data for the bicycle turn signal system, the person doing data collection attaches the data-logger to a friend’s hand as shown in Figure 2 and films the friend making example signals as shown on the left in Figure 1. The data-logger is in the orange box attached to the glove.

To synchronize the video and data, the user records on the video a red flash generated by the data-logger ten seconds after the data-logger is turned on. The data-logger does not need to be visible after the red flash is captured. The data-logger logs the time of the first red flash (in milliseconds elapsed since the processor was turned on). At the beginning of labeling in VAT, the user locates and marks the first video frame that contains the first red flash. VAT can then synchronize the video and data for as long as both continue recording. If either is stopped or turned off, synchronization must be repeated.

Label Events in VAT
After marking the frame containing the first red flash, the user is presented with a list of buttons, the video, and the data stream. The buttons are used to label events, typically in the form of event parts as shown in Figure 3. Video playback can be controlled using the space bar to toggle play and pause and the arrow keys to advance a frame forward or backward.

As is common in these kinds of interfaces, the data stream display is synchronized with video playback so that the red vertical line in the data stream display is always over the data recorded during the current video frame. The data stream is also interactive. Clicking in the data stream advances video playback to the corresponding location in the video; scrolling the mouse wheel zooms in and out of the data stream time scale. The numbers beneath the data stream display represent the number of seconds since the data-logger was turned on.

In Figure 3, the user is in the process of labeling the “raise” piece of a left turn event. The orange highlight in the data display window marks the region of data currently labeled “raise.” The highlight color corresponds to the color of the label shown under the “Left Turn” button. We will refer to such periods of time (sequences of measurements corresponding to some event in the data) as “event regions”.

In the frame of video shown in Figure 3, the rider has extended his arm to the left and will hold it stationary for about one second (from second 55 through second 56) and then put his hand back on the bicycle handlebars. The video can be helpful in guiding non-experts to find event boundaries.

The annotated data (but not the video) is passed to a machine learning algorithm. We used as Gaussian-mixture model based hidden Markov model (HMM-GMM). HMMs and other sequence models have been applied to activity recognition [2, 13].

Results

We investigate the efficacy of non-expert data collection and annotation in four ways, each supported by user studies: first we assess the consistency and accuracy of labels against each other and relative to a gold standard; second, we demonstrate that, when properly supported by software and other tools, novice users can collect physical event data suitable for annotation; third, we assess the relative quality of event classifiers obtained using those labels; and fourth we build functioning prototype devices using labels from non-expert annotators as described in this paper.

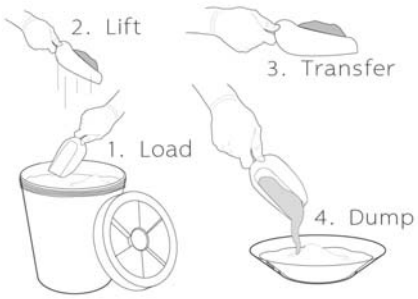

Figure 4: Four pieces of a scoop event: loading, lifting, transferring and dumping.

Consistent and Accurate Labeling

Ten participants (5 male, 5 female) were recruited from an introductory programming class. All were full-time college students. Although they were recruited from an introductory programming class we selected only students with non-STEM majors. Participants received $30 US for their participation. Participants were taught the basic process by being guided through an example annotation task involving throwing and catching a Frisbee. This is similar to a brief training video or short training task that might need to be given to a crowd worker before they begin work.

Participants were then shown video of a person walking with a cane and data collected by the data-logger. The data-logger was attached to the cane. The participants were asked to label a "step" event that was broken into two pieces: (1) the part of the step where the cane tip is in contact with the ground and (2) the part of the step where the cane is not in contact with the ground. The video included 9 steps to be annotated over less than 4 minutes of action. The video and data include motion other than walking, such as picking up the cane, in addition to the steps to label.

First, we compared participant annotations to each other. This simulates the case in which there is no correct reference annotation set created by an expert as might arise when very large data sets are collected via the crowd and it is too expensive to have experts annotate the data.

We did this evaluation using Krippendorff's alpha coefficient (henceforth "α"), a common statistical measure of annotator agreement [10]. We also experimented with the more limited Cohen's Kappa statistic [3], similar results are obtained when it is used. α is typically used in content analysis where textual units are categorized by annotators, which is very similar to the labeling of event pieces we asked our annotators to do. α is well suited to the evaluation of raw annotations, each annotation being a single label for each sample in the data, that is, for each time step. The coefficient is a single scalar value which is insensitive to the numbers of annotators, small sample sizes, unequal sample sizes and missing data.

The α for our group of annotators compared to each other is 0.95, which is quite high; values above 0.80 are considered reliable [10].

Second, we compared participant annotations to a correct set of annotations generated by a three members of the research team. While the previous comparison determined that non-expert annotators agree with each other, this comparison determines the degree to which non-experts agree with experts. This is important because it indicates the degree to which non-experts can be expected to replace experts in crowd sourced annotation tasks.

Participant	α
P1	0.933
P2	0.959
P3	0.727
P4	0.962
P5	0.977
P6	0.939
P7	0.939
P8	0.965
P9	0.975
P10	0.977
avg.	0.935

Table 1: Table of α-values for labeling events in the cane data set when compared to a gold standard. All annotators except one were reliably in agreement with the gold standard.

Three members of the research team annotated the data using the a-b-arbitrate [8] approach to create a correct "gold standard" annotation. In the a-b-arbitrate approach, annotators "a" and "b" make selections and the arbitrator breaks ties if needed. In our case there was only one tie (one video frame) which had to be broken by the arbitrator. We then computed the pair-wise α for each study participant and the gold-standard data. The average of the pair-wise α coefficients is 0.935 with standard deviation 0.073. These pairwise αvalues are shown in Table 1. Note that annotator 3 (α = 0.727) seems to have gotten confused part way through the task; the last few annotations were exactly opposite what they should have been, that is consistently cane down when it should have been cane up. Future work in annotation might consider strategies to prevent this sort of error. We did not remove this participant from the statistics given here since this is surely the sort of error novices (and experts) might make. Note also that even annotator 3's value is in the range where tentative conclusions can be drawn ($0.800 \geq \alpha \geq 0.667$) [10].

Gathering Data for Annotation

To assess non-experts ability to collect, rather than just annotate, data and video we asked the same 10 participants to collect data and video for a scoop counting application shown in Figure 4. The design of the scoop counter is similar to other systems in which accelerometers attached to cooking utensils are used to detect events [15, 14].

Participants were given a consumer grade video camera and a data-logger already attached to the scoop. Participants then gather scoop video and data. They then demonstrated that this data is suitable for annotation by annotating his or her own data.

In this study, we evaluated the performance of the HMM-GMM in learning a classifier on data collected and annotated by non-experts.

Quality of Event Classifier

Using the scoop annotations obtained as described above, we trained an HMM-GMM model based on the description in [17] and measured the quality of the event classifier for each participant. (Because participants each labeled different video and data, we cannot compute α to measure annotator reliability for this group of annotations.)

We could have measured α for the learned event classifier, but instead we use a metric more suited for assessing the event classifier as part of an interactive system. Our intended applications involve the construction of interactive physical objects. In that context what matters most is identifying events quickly enough and accurately enough to build interactive systems.

Given that objective, we deemed the trained learner acceptable if it could detect the completion of an event with the period 0.5 seconds early or 0.5 seconds late relative to the actual occurrence of the event (henceforth "β"). We use the same a-b-arbitrate process [8] described in Section to identify the actual event boundaries.

We computed recall, precision and the F1 metric using β by counting true positives, false negatives, and false positives as follows: A true positive is an event that is inferred within 0.5 seconds of a true event, a false positive is an event that is inferred by the learner near no true event or near a true event that has already been accounted for by some other, closer, inferred event, and a false negative is a true event that occurs with no inferred event within 0.5 seconds.

First, we consider the quality of an event classifier for recognizing scoop events from a single participant. This important for understanding the learning of "personal"

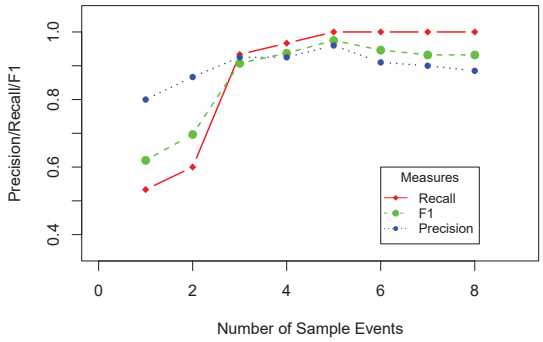

Figure 5: Average learner performance as a function of the number of event regions labeled. Labeling more than 5 or 6 event regions produces little improvement in the learned event classifier in this setting. The learner was trained and evaluated on data from a single participant at a time.

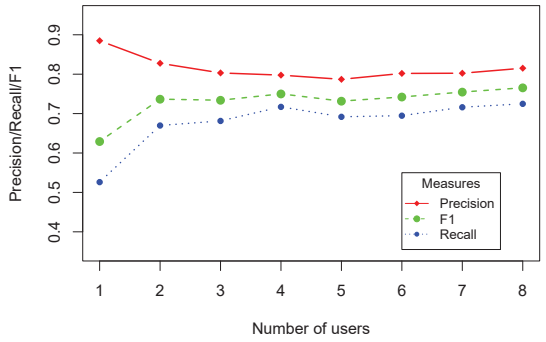

Figure 6: Average prediction performance as a function of the number of participants' data used in training. The test set was obtained from three people who's data was not included in the training set. Learning a event classifier from data collected by 2-4 people results in a noticeably more general event classifier.

classifiers which are tuned to a single person. We collected 12 event regions and annotations from each participant. We used 9 as training data and 3 as test data in exhaustive cross-fold validation. Recall as shown in Figure 3 by "event region" we mean a sequence of measurements (a period of time) that are all part of the same event; thus while our participants perceive these as a small number of labeled events, the machine learning algorithm actually processes them as many labeled time-steps, in a way similar to Image Processing with Crayons [5]. Event classifiers learned from the participants' data and labels achieved an average precision of 0.91 ($sd = 0.15$), average recall of 1 ($sd = 0.0$) and average F1 of 0.92 ($sd = 0.093$).

Figure 5 illustrates the sensitivity of the learner to the number of event regions used as training data. The vertical axis shows the average precision, recall, and F1 score for the 10 participants, and the horizontal axis shows the number of labeled event regions used as input. For this data set, learning slows at 6 event regions, suggesting that the payoff associated with labeling more than 6 event regions is minimal for this data and for our HMM-GMM algorithm.

Next, we explored how well the classifier generalizes on data generated by *other* participants. This is important for crowd-sourcing collection of many examples of an activity and using all of them to learn a single classifier that activity.

Figure 6 shows how the learner generalized. The vertical axis in Figure 6 shows the average performance of the learner on scoop data generated by three people who did

not participate in the study. As before, ground truth event labels for these three samples were generated using a-b-arbitrate. The horizontal axis shows the number of participants' data used to learn the event classifier. For example, "2" on the horizontal axis means that we combined the scoop data from 2 participants to train a event classifier, then used that event classifier on the amalgamated remaining scoop data. The value shown for "2" is the average of this process over all possible sets of 2 annotators.

Collecting data from 2 people lead to a more general event classifier than collecting data from 1 person, as expected. Collecting data from more than 2 people led to slight improvements in generality on this data set. Further work is needed to scale the study to data collected by hundreds of people.

Building Interactive Systems

The third study involved building interactive systems using data collected and annotated by non-experts. A total of 6 people participated in data collection and analysis but a collection of experts built the actual systems.

The purpose of the third study is to demonstrate that non-experts can provide annotations suitable for building working prototype devices. We built three devices: a cooking scoop counter, a bicycle hand signal and a medication reminder.

The cooking scoop counter announced the number of scoops used as the user scoops an ingredient (like flour or sugar) into a bowl. The purpose of the scoop counter is to help a cook remember how much of an ingredient they have added. For this application we used data and annotations collected as part of the scoop data study described above.

The bicycle hand signal system converts standard bicycle hand signals into LED signals on a backpack as shown in Figure 1. The data for this example were collected and annotated by a design student working with an assistant. While the design student brought significant fabrication expertise to the problem, the student did not have expertise in data collection or annotation.

The final application was a medication reminder designed to remind a patient to take their medicine and notify a caretaker when it has been taken, similar to the one constructed by Dey et al. [4]. To build the system, we attached a data-logger to the side of an empty pill bottle, filled the bottle with candy, and collected sample data for three different users. Data for this system were collected and annotated by 4 university students with design backgrounds (3 male, 1 female) to collect the data required to train the event classifier while recording the time required to complete each step. Participants each required between 59 and 80 minutes, with an average of 71 minutes, to collect and label the video and data. A little more than half of that time, 39 minutes, was spent labeling pill taking events in the data.

Conclusion and Discussion

Our results suggest that non-expert users can collect and annotate human activity data synchronized with video, and that an HMM-GMM algorithm can learn reasonably accurate and general event classifiers from these data and annotations. Users labeled events in the cane dataset with very good inter-rater reliability as measured by a Krippendorff's alpha coefficient of 0.935. An HMM-GMM algorithm learned reasonably accurate event classifiers from user-generated labels in the scoop dataset with an average F1 score of 0.92, which is sufficient for prototypes and research projects. More importantly, data and

annotations from as few as 2 novices generalized well to the data produced by others. We built three functioning systems from non-expert data and annotations.

One application of our results is that crowd-sourced data collection and annotation of both video and data is feasible for simple tasks involving a single sensor. It may be feasible to carry crowd-sourced collection of large data sets for common activities. It may also be feasible to ask a single person to collect, annotate and build a classifier from their activities for a kind of personalized activity recognizer.

A weakness of our results is that they were collected using a single data collection and annotation scheme and that they were used in a single machine learning algorithm. Studies involving a different scheme or algorithm may lead to different results. Exploration of the resilience of various machine learning algorithms to novice annotation is also deferred to future work.

Another weakness of our results is that we consider only limited classes of data and annotation. For example, our data involve a single sensor and a limited set of activity labels. Richer data sets involving more sensors and more activity labels may be prohibitively difficult for non-experts to collect and annotate.

REFERENCES

1. Abdelkareem Bedri, Apoorva Verlekar, Edison Thomaz, Valerie Avva, and Thad Starner. 2015. Detecting Mastication: A Wearable Approach. In *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction (ICMI '15)*. ACM, New York, NY, USA, 247–250. DOI: http://dx.doi.org/10.1145/2818346.2820767
2. Andreas Bulling, Ulf Blanke, and Bernt Schiele. 2014. A tutorial on human activity recognition using body-worn inertial sensors. *ACM Computing Surveys (CSUR)* 46, 3 (2014), 33.
3. J. Cohen. 1960. A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement* 20 (1960), 37–46.
4. Anind K. Dey, Raffay Hamid, Chris Beckmann, Ian Li, and Daniel Hsu. 2004. A CAPpella: Programming by Demonstration of Context-aware Applications. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '04)*. ACM, New York, NY, USA, 33–40. DOI: http://dx.doi.org/10.1145/985692.985697
5. Jerry Alan Fails and Dan R. Olsen Jr. 2003. Interactive machine learning. In *Proceedings of the 8th international conference on Intelligent user interfaces*. ACM, 39–45.
6. Adam Fouse, Nadir Weibel, Edwin Hutchins, and James D. Hollan. 2011. ChronoViz: A System for Supporting Navigation of Time-coded Data. In *CHI '11 Extended Abstracts on Human Factors in Computing Systems (CHI EA '11)*. ACM, New York, NY, USA, 299–304. DOI: http://dx.doi.org/10.1145/1979742.1979706
7. Benjamin H. Groh, Frank Warschun, Martin Deininger, Thomas Kautz, Christine Martindale, and Bjoern M. Eskofier. 2017. Automated Ski Velocity and Jump Length Determination in Ski Jumping Based on Unobtrusive and Wearable Sensors. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 1, 3, Article 53 (Sept. 2017), 17 pages. DOI: http://dx.doi.org/10.1145/3130918
8. Derek L Hansen, Patrick J Schone, Douglas Corey, Matthew Reid, and Jake Gehring. 2013. Quality control

mechanisms for crowdsourcing: peer review, arbitration, & expertise at familysearch indexing. In *Proceedings of the 2013 conference on Computer supported cooperative work*. ACM, 649–660.

9. Michael Kipp. 2001. ANVIL - A Generic Annotation Tool for Multimodal Dialogue. (2001).
10. Klaus Krippendorff. 2012. *Content analysis: An introduction to its methodology*. Sage.
11. Walter S. Lasecki, Young Chol Song, Henry Kautz, and Jeffrey P. Bigham. 2013a. Real-time Crowd Labeling for Deployable Activity Recognition. In *Proceedings of the 2013 Conference on Computer Supported Cooperative Work (CSCW '13)*. ACM, New York, NY, USA, 1203–1212. DOI: http://dx.doi.org/10.1145/2441776.2441912
12. Walter S. Lasecki, Young Chol Song, Henry Kautz, and Jeffrey P. Bigham. 2013b. Real-time Crowd Labeling for Deployable Activity Recognition. In *Proceedings of the 2013 Conference on Computer Supported Cooperative Work (CSCW '13)*. ACM, New York, NY, USA, 1203–1212. DOI: http://dx.doi.org/10.1145/2441776.2441912
13. Donald J Patterson, Dieter Fox, Henry Kautz, and Matthai Philipose. 2005. Fine-grained activity recognition by aggregating abstract object usage. In *Wearable Computers, 2005. Proceedings. Ninth IEEE International Symposium on*. IEEE, 44–51.
14. Cuong Pham and Patrick Olivier. 2009. *Ambient Intelligence: European Conference, AmI 2009, Salzburg, Austria, November 18-21, 2009. Proceedings*. Springer Berlin Heidelberg, Berlin, Heidelberg, Chapter Slice & Dice: Recognizing Food Preparation Activities Using Embedded Accelerometers, 34–43. DOI: http://dx.doi.org/10.1007/978-3-642-05408-2_4
15. Thomas Plötz, Paula Moynihan, Cuong Pham, and Patrick Olivier. 2011. Activity recognition and healthier food preparation. In *Activity Recognition in Pervasive Intelligent Environments*. Springer, 313–329.
16. Peter Wittenburg, Hennie Brugman, Albert Russel, Alex Klassmann, and Han Sloetjes. 2006. ELAN: a professional framework for multimodality research. In *5th International Conference on Language Resources and Evaluation (LREC 2006)*. 1556–1559.
17. Steve Young, Gunnar Evermann, Mark Gales, Thomas Hain, Dan Kershaw, Xunying Liu, Gareth Moore, Julian Odell, Dave Ollason, Dan Povey, and others. 1997. *The HTK book*. Vol. 2. Entropic Cambridge Research Laboratory Cambridge.