Searching Recorded Speech Based on the Temporal Extent of Topic Labels

Douglas W. Oard* **and Anton Leuski**
University of Southern California Information Sciences Institute
4676 Admiralty Way, Suite 1001
Marina Del Rey, CA 90292-6695
oard@glue.umd.edu, leuski@isi.edu

Abstract

Recorded speech poses unusual challenges for the design of interactive end-user search systems. Automatic speech recognition is sufficiently accurate to support the automated components of interactive search systems in some applications, but finding useful recordings among those nominated by the system can be difficult because listening to audio is time consuming and because recognition errors and speech disfluencies make it difficult to mitigate that effect by skimming automatic transcripts. Support for rapid browsing based on supervised learning for automatic classification has shown promise, however, and a segment-then-label framework has emerged as the dominant paradigm for applying that technique to news broadcasts. This paper argues for a more general framework, which we call an activation matrix, that provides a flexible representation for the mapping between labels and time. Three approaches to the generation of activation matrices are briefly described, with the main focus of the paper then being the use of activation matrices to support search and selection in interactive systems.

Introduction

Recorded speech is a linear medium in which rapid skimming is hard to support. So although search based on speech recognition can be efficient, selection from a set of retrieved recordings, each of which might be several hours long, would be a time consuming process. Two approaches to this challenge have emerged: passage retrieval, and visualization. The key idea in passage retrieval is to divide speech recognition transcripts into thematically coherent (and relatively brief) segments, assign a score to each segment, and then present metadata that describes each high-scoring segment in the result list (Wechsler & Schaüble 1995). Visualization-based approaches, by contrast, retain the integrity of the original recording, but suggest points at which to begin the replay, typically using a timeline visualization. These suggestions can be based on thematically coherent segments that indicate both the onset and the temporal extent of potentially useful segments (as in the AT&T SCAN system (Whittaker *et al.* 1999)) or they can indicate only the onset (as in the HP Labs SpeechBot system (Thong *et al.* 2000)).

All three of the examples cited above are based on the presence of terms (words and/or phrases) that occur in the speech recognition transcript. In this paper, we explore alternative techniques in which automatic text classification is used to label periods of time in the recording, with a greater degree of abstraction than would be possible using just the actual words that were spoken. Classification-based approaches seem particularly well suited for use with linear media, since topic labels are easily skimmed. For example, Merlino and Maybury found that selection decisions could be made more than twice as quickly and with comparable accuracy for news broadcasts when based on automatically assigned topic labels than when based on closed-caption text (precision increased 19%, although recall decreased by 11%) (Merlino & Maybury 1999).

Segmentation and topic labeling have been the focus of the Topic Detection and Tracking evaluations, in which technology for improving access to audio and print news sources has been assessed annually since 1998 (Wayne 2000). In the TDT evaluations, segmentation and classification are modeled as separable problems, in which the system first seeks to detect segment boundaries created during the production process, and then seeks to label each segment with one or more topics. The BBN Oasis system incorporates this idea, displaying topic labels adjacent to the speech recognition transcript. This decomposition is appropriate for broadcast news, where segment combination is a natural part of the editorial process by which a news broadcast is created. The utility of such a decomposition for naturally produced speech is not as clear clear for two reasons:

- Spontaneous speech may contain digressions that would prevent the construction of contiguous segments.
- Searchers with different goals might seek segments with different topic granularity (e.g., a teacher might want a fairly broad self-contextualizing segment, while a documentary film maker might need only a brief statement to illustrate a single concept).

These challenges could be accommodated by constructing a

*Permanent address: College of Information Studies and Institute for Advanced Computer Studies, University of Maryland, College Park, MD 20742

multitude of brief segments and then defining relationships between those (e.g., topic containment and topic threading) that would support the expected searches. We believe, however, that such an approach would lack elegance and limits generality, however. In this paper, we propose an alternative approach to the generation and use of topic labels in which the span of each label is determined individually. We begin with a description of the representation, then show how this representation can be used to support search and browsing. The paper concludes with a few remarks on the nature of the problems to which our proposed technique might productively be applied.

Activation Matrices

We begin by defining an abstract representation for the span of each label. Figure 1 illustrates this structure, which we call an "activation matrix." The rows represent a set of topic labels, for which some relationships between the labels may be known. In this paper we will restrict our attention to the case in which the relationships form one or more hierarchies, such as might be commonly found in a thesaurus or an ontology. The columns represent units of time, with the granularity of the temporal representation left unspecified at this point as an implementation detail. Since two words per second would be relatively fast speech, it seems unlikely that sub-second temporal resolution would be needed in any application. So it might be helpful to think of each cell as representing one second of time. Each cell of an activation matrix contains a single real-valued number that represents the likelihood that the label represented by the row should be assigned at the time represented by the column. The precise nature of these likelihood values is also left unspecified as an implementation detail, but for the remainder of this paper we will choose to think of them as probabilities. In Figure 1, the degree of shading in a box is intended to represent the probability that the row's label should be assigned at the column's time, with the darkest shades representing certainty.

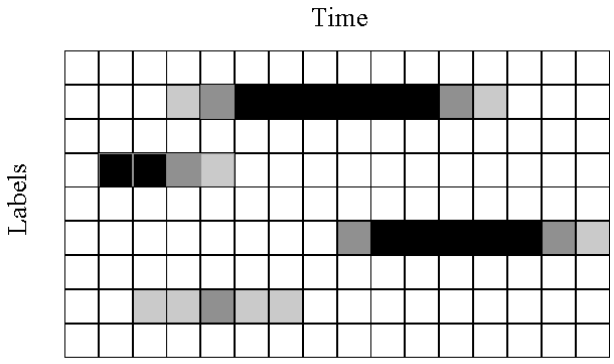

Figure 1: A sample activation matrix.

The first natural question to ask is where such activation matrices might come from. One possibility is that they might result from manual annotation of a collection. For example, professional catalogers might listen to a recording and mark the onset and termination of discussion of topics from a predefined topic inventory. Another way in which an activation matrix might be created is through a cascaded segment-then-label process using techniques demonstrated in the TDT evaluations. Figure 2 illustrates a third way in which such matrices might be created, relying on hand-labeled training data to train a model for the annotation process, and then applying that model to future data. Words contained in the speech recognition transcript are one example of a feature sequence that might be useful in such a classifier. For example, we might build a language model for each topic from the training data, and then assign probabilities to each topic in the activation matrix for previously unseen data based on the degree of match to the language model associated with each row in the matrix. Other features (e.g., turn-taking behavior and silence) may also prove valuable in some applications. Because our goal in this paper is to focus on how such activation matrices might be used, we will leave the detailed design of the model for future work (another of the "implementation details").

In most applications that we envision, the activation matrix would be quite sparse, with only a small fraction of the labels active at any particular time. Some form of compact representation would therefore be possible, and indeed would be necessary if we are to efficiently store any activation matrix with more than a few rows. Some variant of the inverted file index structure used in information retrieval would seem to be appropriate. For example, we could model the pattern of probabilities as a set of linear regions, recording the start time, initial value, and slope of each region. Each row could then be walked from left to right, easily reconstructing the value in any cell. Since it seems unlikely that appropriately smoothed probability values would exhibit high-frequency jitter in a real application, such a coding scheme would likely be both relatively compact and easily searched.

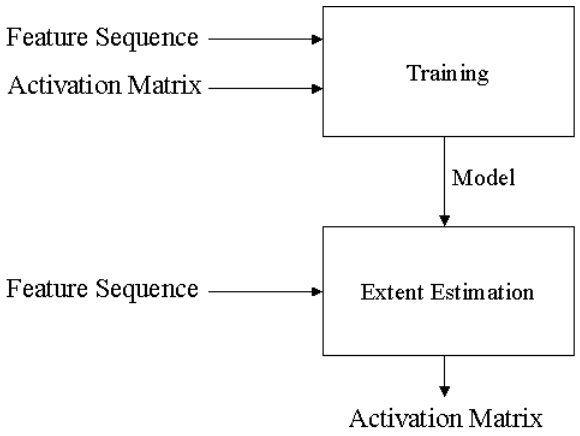

Figure 2: Supervised learning for activation matrices.

Finally, the question of what constitutes a good activation matrix is also important. For intrinsic evaluation, we could

imagine a process in which some "gold standard" activation matrices are created by hand and then compared with those that are automatically generated. One possibility would be to use the L_1 norm of the matrix difference:

$$L_1 = \sum_{i,j} |M_{i,j} - A_{i,j}| \quad (1)$$

where $M_{i,j}$ is the probability at row i and column j of the manually created gold standard and $A_{i,j}$ is the corresponding value for an automatically generated activation matrix.

The effect of this measure is to measure the average difference between the predicted probability that a topic should be active at a specific time and a corresponding ground truth value for that probability. For reasons of economy, the ground truth is likely to be binary (i.e., a human would judge whether the topic label should be present or absent). The absolute value operator gives equal weight to errors in either direction; biased versions of this measure might also be used if, for example, precision were thought to be more important than recall in some application. It might also be useful to normalize the measure in some way to facilitate interpretation. But the fundamental limitation of any intrinsic measure is that it can only reflect how well what is produced matches what is thought to be needed. To assess how well an activation matrix actually supports some task, we must examine how it will be used. That is the focus of the next two sections.

Searching Activation Matrices

Modern text retrieval systems represent the contents of a document collection in a manner very similar to that of an activation matrix. In so-called "natural language" text retrieval, the rows typically represent terms (words or phrases), the columns typically represent documents or (for "passage retrieval") segments of documents, and the elements typically represent the degree to which a term describes a document (or segment). Mapping these concepts to an activation matrix is direct, with labels filling the role of terms, the segments possibly being shorter than is typical in text retrieval, and the degree of description being represented as a probability. We can therefore build on the substantial body of work on the design of text retrieval systems (Baeza-Yates & Ribeiro-Neto 1999).

There are two fundamental approaches to information retrieval: exact match retrieval and ranked retrieval. If the activation matrix contains only binary-valued elements (recording the presence or absence of a label at a time), then Boolean logic can be used in the query language to allow any possible combination of active labels to be specified. The natural result would be a set of contiguous spans in which the specified combination of labels is active. This set might further be ranked (e.g., in order of decreasing duration of the retrieved span) or clustered (e.g., with all spans from the same recording being shown on a single timeline).

Boolean logic offers an expressive query language, to which additional capabilities can be added using proximity operators and thesaurus-based query expansion. But Boolean logic has two key limitations that are important in practice. First, effectively expressing a query using Boolean logic requires a good deal of expertise (knowledge of Boolean logic, familiarity with appropriate iterative search strategies that minimize the all-or-nothing problem associated with overly general and overly specific queries, and a sufficient understanding of collection characteristics). An alternative approach is to use somewhat less expressive "natural language" queries to identify a broad range of potentially useful documents, emphasizing support for browsing by displaying the retrieved set in order of decreasing probability (or degree) of topical relevance to the query. The probabilities in the activation matrix offer a natural basis for creating such a ranked list.

Ranked retrieval systems typically compute a score for each segment as follows:

$$s_j = \sum_i f(w_{i,j}, c_i) \quad (2)$$

where s_j is the score assigned to segment j, $w_{i,j}$ is the degree to which label i describes segment j and c_i is the number of segments that are described by label i. The function $f(\cdot,\cdot)$ typically increases monotonically with w and decreases monotonically with c, capturing the intuition that relatively uncommon terms offer the most useful basis for ranked retrieval. One well known variant of equation (2) is the classic *tf*idf* formula:

$$f(w_{i,j}, c_i) = w_{i,j} * \log((N/c_i) + 1),$$

and some more recently developed variants such as the Okapi measure (Robertson *et al.* 1994) are now also widely used.

Some systems combine ranked retrieval techniques with Boolean methods to provide expressive query languages that also include a "relevance ranking" capability. For example, modern Web search engines now often perform an implicit "AND" operation across the searcher's query terms and then ranked the returned results using some variant of (equation 2). The optimal balance between query complexity and interactive selection depends, of course, on how well we can support the process of browsing retrieved sets. That is the focus of the next section.

Visualizing Activation Matrices

Timelines are an attractive choice as a visualization for linear media because they leverage a familiar metaphor for depicting temporal relationships. For example, SpeechBot provides a single timeline on which it places the time offsets of the good query matches; clicking on any of those points then initiates replay. Working with text documents, Hearst introduced a segmentation-based timeline-like visualization, shading each document segment in proportion to the density of query terms found there (Hearst 1995). Byrd introduced a segmentation-free variant of this idea, closely coupling control and display by marking the scrollbar with different colors to indicate the presence of individual query terms at that point in a document (Byrd 1999). The AT&T SCAN system applied these ideas to recorded speech, depicting segment boundaries on a timeline, and then using color to indicate which query terms are present in each segment.

One natural way to extend these ideas to accommodate labels with possibly overlapping scopes is to use multiple timelines. For example, the Jabber system uses this technique to visualize speaker activity at meetings (Kazman *et al*. 1996).

The OntoLog system adopts a similar approach showing the span of labels assigned by an end user performing video analysis (Heggland 2002). Because the number of available labels may exceed that which can be depicted using the available screen space, OntoLog incorporates facilities for hierarchically aggregating timelines. OntoLog includes a tree browsing function for expanding and collapsing branches in a predefined hierarchy in a manner similar to a file system browser. The tree browsing process is closely coupled with timeline aggregation so that the timeline for each label is displayed immediately to the right of its position in the tree browser. Timelines associated with leaf nodes in the tree are shown as thin lines of uniform width that may be present or absent at any time; aggregated timelines associated with a collapsed interior node have a variable thickness depending on the number of subsumed leaf node labels that are active at each point in time.

Figure 3 shows one way in which these ideas might be extended to present the information available in activation matrices. It illustrates three key extensions:

1. Using grayscale shading to depict probabilities (when the activation matrix contains more than just binary values).
2. Providing the user with control over the display order for individual topic timelines by decoupling the display order from from the layout of the tree browser that is used to select terms from the thesaurus. We expect this to be useful when the thesaurus contains a a large number of possible labels.
3. Closely coupling timeline with the display of associated text. This associated text may be a script (e.g., for a speech), a manually-prepared edited transcript (e.g., for an oral history interview), the output of an automatic speech recognition (as is used in SpeechBot), or some other form of time-aligned summaries. Since lines of text are normally read from top to bottom (in Western languages), we use a set of vertical rather than horizontal timelines. This makes it possible to naturally couple control and display in a manner similar to that introduced by Byrd.

Essentially, we visualize the activation matrix by showing which labels are associated with each time moment. The total number of labels in the activation matrix could be far larger than could possibly be displayed simultaneously—perhaps on the order of tens of thousands in some applications. Thus the task of the interface breaks into two subtasks: (1) select an interesting subset of the labels and (2) use this subset to navigate the media stream. In this section we consider how a chosen set of labels can be used to describe the stream and then we discuss how the labels can be selected.

Figure 3 shows a visual mockup of our notional interface for a real interview (from the NASA oral history collection). The main part of the window is taken by a scrolling view

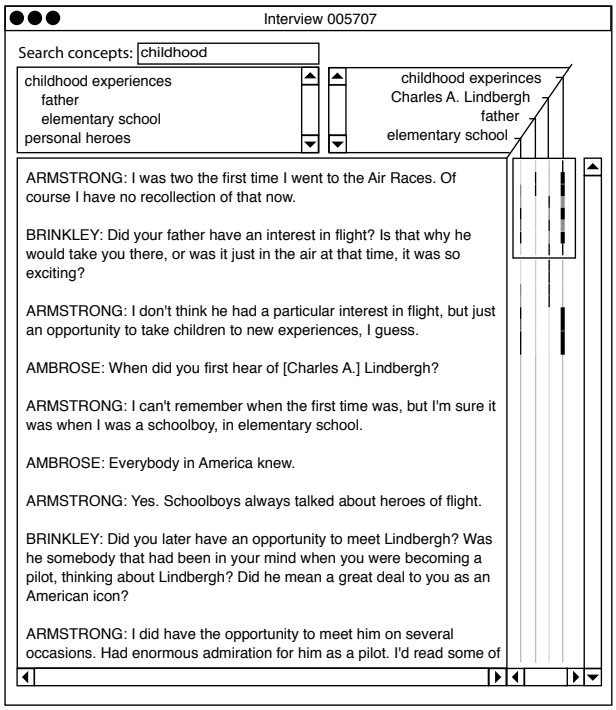

Figure 3: A mockup of the interface.

of an manually-prepared edited transcript. The top right region of the window shows four labels that we have selected for illustrative purposes: "childhood experiences," "Charles A. Lindbergh," "father," and "elementary school." Each label has an associated (vertical) timeline along the right edge of the display that is joined to the (horizontal) label by an elbow-shaped line.

The black rectangle in the top right quarter of the timeline area depicts the portion of the recording for which the associated text is presently visible. This box can be manipulated in the same manner as a scrollbar, thus rapidly recentering the text view on an area in which an interesting combination of the selected labels is likely to be active.

We use shades of gray to depict the probabilities found in the activation matrix, with higher probabilities being represented with a darker shade. For example, the label "father" is very likely to be present in the second speaker turn (since the corresponding points in the timeline near the top of the rectangle are very dark), while that label is unlikely to be present in the first speaker turn since the timeline is very light at the very top of the rectangle.

In our example, "childhood experiences" labels an interior node in the hierarchy, with "father" and "elementary school" labeling its leaf nodes. The corresponding timeline (the rightmost one) is often depicted as somewhat thicker than the timelines for the child topics. The thickness of a timeline for an interior node at any time is related to the number of subordinate leaf nodes that are active at that time, while the shading of the timeline for an interior node is related to the cumulative probability that some subordinate leaf node is active at that time. For example, the fifth speaker

turn describes elementary school experiences and therefore both the "elementary school" and the "childhood experiences" timelines are darkened. The first speaker turn also describes childhood experiences but not elementary school, so the "childhood experiences" label is thinner at that point.

We now return to the question of how these labels were selected. The top left portion of the window is intended to show a tree browser similar to those used for file system browsing in most popular operating systems. Widgets (not shown due to the limited fidelity of this mockup) are used to expand and collapse the display of subordinate labels for interior nodes, and the scrollbar at the right can be used to recenter the display on another part of the hierarchy. There is also a search function at the top of the window to augment tree browsing with a capability to search for labels based on the words that they contain. Once an interesting topic is found, the user can then drag the label to an appropriate point in the top right region. That action will result in reordering the list of labels selected for display and will create a new timeline for the selected label.

When working with relatively small lists in the upper right region, users may wish to reorder the list manually by dragging labels up and down the list. For larger lists, we might explore automatic reordering techniques, perhaps placing the labels that are most often active in this recording at the top of the list. When more labels are selected than can be displayed simultaneously in the list at the upper right, the user can scroll through the list using either scrollbar to the right of the labels or the scrollbar below the timelines. Either action will bring both the labels and their associated timelines into view.

Our example here is meant to be illustrative rather than comprehensive. But it does serve to reinforce our key point: that visualizing activation matrices could be a useful way of navigating through a large recording. In this case, the user can see at a glance that it is the early portion of the recording where Armstrong describes his childhood experiences, and that he also describes his relationship with Charles Lindbergh in that same region. Such a capability would be more difficult to represent in a natural manner when using a segment-then-label approach, since it is not clear how a segment relating to Lindbergh would relate to a segment describing childhood experiences. This example also serves to illustrate some of the potential value of using automatically assigned labels rather than spoken words, since, for example, "childhood" is never uttered once by either speaker. This is, however, merely an illustration—much remains to be learned from actually building such a system and then trying it out with real users.

Conclusion

We have presented a general framework that we call activation matrices for labeling topics in recorded speech in a manner that accommodates both uncertainty and the possibility that the span of some labels may overlap in ways that are not synchronized. We have suggested three ways in which activation matrices might be generated and have sketched out how they might be evaluated and how they might be used by the automated component of search systems and for visualization of search results. When transcripts are available, term-based and label-based techniques offer potential synergies for both search and browsing. In the absence of transcripts, label-based techniques offer the only practical support for these tasks. Activation matrices offer a potentially useful alternative to the present "segment-then-label" approach when searching unedited speech, but several points should be borne in mind:

- As with any label-based technique, an initial investment is necessary to develop an appropriate label inventory and to hand-label some training data.
- Manually assigning separate onset and termination points for each label will likely require more effort than the segment-then-label approach. It may be possible, however, to use relatively short hand-labeled segments as a basis for training a classifier to automatically generate activation matrices for unlabeled data.
- We have not yet tried these ideas with real data.
- Activation matrices are somewhat less expressive than lattices because they do not encode sequential dependencies. It remains an open question whether a topic lattice might be used in some way that would provide even better support than activation matrices for search and/or browsing.

This is a work in progress, and we expect to have some of these ideas better worked out by the time of the symposium. We're looking forward to the opportunity to discuss this approach, and welcome comments from the reviewers on productive directions that we might explore.

Acknowledgments

The authors are grateful for Jon Heggland for helping us to see the link between ontologies and multimedia annotation, and to the MALACH team for their helpful comments on the ideas reported here. This material is based upon work supported by the National Science Foundation (NSF) under grant number 0122466. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF.

References

Baeza-Yates, R., and Ribeiro-Neto, B., eds. 1999. *Modern Information Retrieval*. New York: Addison Wesley.

Byrd, D. 1999. A scrollbar-based visualization for document navigation. In *Proceedings of the Fourth ACM International Conference on Digital Libraries*, 122–129.

Hearst, M. A. 1995. Tilebars: Visualization of term distribution information in full text information access. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, 56–66. ACM.

Heggland, J. 2002. OntoLog: Temporal annotation using ad-hoc ontologies and application profiles. In *Sixth European Conference on Research and Advanced Technology for Digital Libraries*.

Kazman, R.; Al-Halimi, R.; Hunt, W.; and Mantei, M. 1996. Four paradigms for indexing

video conferences. *IEEE Multimedia* 3(1):63–73. http://www.cgl.uwaterloo.ca/Jabber/ieee-mm4.ps.

Merlino, A., and Maybury, M. 1999. An empirical study of the optimal presentation of multimedia summaries of broadcast news. In Mani, I., and Maybury, M., eds., *Automated Text Summarization*.

Robertson, S. E.; Walker, S.; Jones, S.; Hancock-Beaulieu, M. M.; and Gatford, M. 1994. Okapi at TREC-3. In Harman, D. K., ed., *Overview of the Third Text REtrieval Conference (TREC-3)*, 109–126. U.S. National Institute of Standards and Technology. http://trec.nist.gov.

Thong, J.-M. V.; Goddeau, D.; Litvinova, A.; Logan, B.; Moreno, P.; and Swain, M. 2000. SpeechBot:a speech recognition based audio indexing system for the web. In *Sixth RIAO Conference on Computer Assisted Information Retrieval*.

Wayne, C. L. 2000. Multilingual topic detection and tracking: Successful research enabled by corpora and evaluation. In *Second International Conference on Language Resources and Evaluation*, 1487–1494.

Wechsler, M., and Schaüble, P. 1995. Speech retrieval based on automatic indexing. In *Final Workshop on Multimedia Information Retrieval (MIRO '95)*.

Whittaker, S.; Hirschberg, J.; Choi, J.; Hindle, D.; Periera, F.; and Singhal, A. 1999. SCAN: Designing and evaluating user interfaces to support retrieval from speech archives. In Marti Hearst, F. G., and Tong, R., eds., *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 26–33.