Joke-o-mat HD: Browsing Sitcoms with Human Derived Transcripts

Adam Janin
International Computer
Science Institute
1947 Center Street, Suite 600
Berkeley, CA 94704-1198
janin@icsi.berkeley.edu

Luke Gottlieb
International Computer
Science Institute
1947 Center Street, Suite 600
Berkeley, CA 94704-1198
luke@icsi.berkeley.edu

Gerald Friedland
International Computer
Science Institute
1947 Center Street, Suite 600
Berkeley, CA 94704-1198
fractor@icsi.berkeley.edu

ABSTRACT

Joke-o-mat HD is a system that allows a user to navigate sitcoms (such as *Seinfeld*) by "narrative themes", including scenes, punchlines, and dialog segments. The themes can be filtered by the main actors and by keyword. For example, the user can select to see only punchlines by Kramer that contain the word "armoire". The system infers the narrative themes using segmentation of the audio track into laughter, actors, words, and music. The segmentation can be generated either by an expert annotator, via automatic methods, or by exploiting human derived (HD) "found" data such as fan-generated scripts and closed captions. We demonstrate browsing one episode of *Seinfeld* using all three methods of generating segmentations.

Categories and Subject Descriptors

H5.5 [**Information Interfaces and Presentation**]: Sound and Music Computing—*Signal analysis, synthesis, and processing*; H5.4 [**Information Systems Applications**]: Navigation

General Terms

semantic segmentation

Keywords

acoustic event detection, speaker id, video navigation

1. INTRODUCTION

The following article briefly describes the Joke-o-mat HD demo system[1]. The Joke-o-mat (non-HD) system won the ACM Multimedia Grand Challenge in 2009. We demonstrate browsing an episode of the situation comedy television show *Seinfeld* — other sitcoms (such as *I Love Lucy* and *The Big Bang Theory*) work equally well, and a slightly modified version of Joke-o-mat has also been used to navigate meetings of a research group at the University of California, Berkeley.

Television show are generally segmented by their producers into narrative themes, such as scenes, punchlines, and dialog acts. The Joke-o-mat HD system analyzes markers such as laughter, music, and speaker identity to allow navigation of sitcoms by these thematic segments. The Joke-o-mat HD system (as compared to the non-HD system) has been augmented with filtering by keyword and a process to generate the segmentations from human derived (HD) "found" data such as fan-generated scripts and closed captions. Although integrating video cues would likely improve the system in various ways, we chose to limit ourselves to audio both to show how far one can get using only audio analysis and because of resource and time constraints.

We assume the following use case: The first time a person watches a Seinfeld episode, he or she needs barely any navigation. Unlike other media, such as recorded lectures, sitcoms are designed for entertainment and should hold the attention of the viewer for the entire length of an episode. When a sitcom is watched at later times, however, a user might want to show a very funny scene to a friend, point out and post the sharpest punchline to his or her facebook network, find the joke about the armoire, or even create a home-made YouTube video composed of the most hilarious moments of his or her favorite actor. In order to do this quickly, a navigation interface should support random seek into a video. Although this feature alone makes search for a particular moment in the episode possible, it remains cumbersome, especially because most sitcoms don't follow a single thread of narration. Therefore, we present the user with the basic narrative elements of a sitcom such as the scenes, punchlines, and individual dialog segments on top of a standard video player interface. A per-actor filter helps to search only for elements that contain a certain protagonist. A keyword filter allows narrowing the search to regions containing particular words. The user is now able to selectively skip certain parts and to directly navigate into elements he or she remembers from the past.

2. TECHNICAL APPROACH

The system consists of two main elements: First, a preprocessing and analysis step, and second an online video browser. The preprocessing step consists of an acoustic event detection and speaker identification step and a nar-

[1] The demo is available at:
`http://www.icsi.berkeley.edu/jokeomat`

ACM Multimedia 2010 Firenze, Italy

rative element segmenting step. The online video browser then uses the output of the narrative elements analysis step to present a video navigation interface to the user.

2.1 Expert Annotation

The acoustic events and speaker identities can be generated in several ways. First, an expert can watch an episode and carefully mark who spoke when, what was said, when laughter occurs, when music occurs, etc. This is a quite costly process. In our experience with annotating multiperson meetings, it generally took 20 hours to transcribe a one hour meeting. We have no reason to expect annotation of videos and television to be significantly different. For this work, one (expert) annotator transcribed one episode of *Seinfeld*, entitled *The Soup Nazi*, for speaker identity, laughter, and music. We did not annotate the words, as word transcriptions are the most difficult costly part of annotation. We consider this transcript to be the "gold standard" by which we measure the other methods.

2.2 Automatic Annotation

Another alternative is to use automatic methods to generate the segmentation. This has the advantage of being both faster and cheaper than using an expert. However, accuracy can suffer. For both acoustic event detection and speaker identification, we used a derivative of the ICSI speaker diarization engine [2] used for speaker identification in meetings [1]. To determine the actual words spoken, we used the SRI/ICSI meeting speech recognition system [3]. Although speech from meetings and speech from sitcoms certainly differ, they are similar enough that the automatic systems trained and tuned on meetings appear to work adequately on sitcoms.

Similar to [1], we trained 60 seconds of each of the following classes: Jerry, Kramer, Elaine, and George, male supporting actor, female supporting actor, laughter alone, music alone, and other noise. The speakers were trained with both pure speech and laughter and music-overlapped speech. The audio track of the episode was then segmented into 2.5 s chunks and each of the chunk was classified to belong to any of the above mentioned classes. The result was saved in NIST-standardized rttm-format (see [2]). When scored against the "gold standard" expert annotation, we obtained a Diarization Error Rate of 23.3%[2]. We would expect the system to perform much better if given more training data. Also, exact annotation of all supporting actors would probably improved the system as well.

Since, as explained above, we did not transcribe the words spoken using expert annotation, we cannot compare the speech recognition results directly. However, anecdotally, the results are reasonably consistent with speech recognition under mismatched training conditions from a far-field microphone (i.e. in the 50% word error range). Note, however, that we do not use the so-called "one-best" results from the recognizer for our keyword filter; rather, we use all the hypotheses that the recognizer produces. Typically, using all the hypotheses allows much higher recognition error with no perceptible drop in performance for end-users of a search or information retrieval system.

2.3 Human Derived Annotation

In addition to using expert annotation or automatic methods, we also used a newly available resource we name "fan-sourced" data. For *Seinfeld*, this consists of scripts and closed captions generated by the extensive fan community. For this work, we used the first well-formed scripts and closed captions we came across from a Google search. We did not select for accuracy.

The scripts all consisted of highly accurate speaker attributed text, but varied extensively in formatting. A significant fraction of the time required to process the fan-sourced data consisted of normalizing the scripts into a machine readable format. An example excerpt from a script can be seen in Figure 1.

```
JERRY: I don't know. Uh, it must be love.
At Monks
========
PATRICE: What did I do?
GEORGE: Nothing. It's not you. It's me. I have a
  fear of commitment. I don't know how to love.
PATRICE: You hate my earrings, don't you?
```

Figure 1: Example of a Fan-sourced Script for *Seinfeld*.

The closed captions all appear to have been generated with *SubRip*, an open source optical character recognition (OCR) program designed specifically for extracting closed captions. The program requires fairly extensive setup and training for accurate use, and as a result, the files provided by two different fans will not be identical. Since the closed captions are generated with a program, the format is quite uniform. This makes processing of the closed captions significantly easier than with the scripts. However, there was one quite interesting problem, almost certainly the result of OCR errors — lower case "L" being used where capital "I" was intended. For example, "lf it makes you happy" instead of "If it makes you happy". Fortunately, this is easy to correct automatically. An example excerpt can be seen in Figure 2.

```
00:04:52,691 --> 00:04:54,716
I don't know. It must be love.

00:05:04,136 --> 00:05:06,468
-What did I do?
-Nothing. It isn't you.

00:05:06,639 --> 00:05:10,598
It's me. I have a fear of commitment.

00:05:10,776 --> 00:05:13,677
-I don't know how to love.
-You hate my earrings, don't you?
```

Figure 2: Example of Fan-sourced Closed Captions for *Seinfeld*.

Neither the scripts nor the closed captions alone are sufficient to generate the segmentations. The scripts lack any

[2] Direct comparisons with [1] and [2] are difficult because we did not use forced alignment to tighten the bounds of the segments.

time information, and the closed captions lack speaker attribution. To generate the segmentations from the scripts and the closed captions, we made two assumptions. First, we assume that the scripts are accurate, both in the words and in the speaker attribution. Second, we assume that the closed captions are correct with respect to time. Since the closed captions appear to be less accurate than the scripts, we do not assume that the words in the closed caption are correct, merely that they are close enough to the script to allow integration of the two sources of data. The actual process to generate the segmentations from the scripts and closed captions is complex, and is presented here only in outline.

First, the scripts and closed captions are normalized for spelling, capitalization, hyphenation, discourse markers (e.g. ahh, mm-hmm), etc. Next, the scripts and closed captions are optimally aligned using minimal edit distance on the words. If the segments' start and end agree (e.g. "I don't know. Uh, it must be love."), then we have a start time, end time, speaker attribution, and word sequence for a segment. We use a speech recognizer to determine the start and end time of each word within the segment. Often, however, the start and end of the segments do not agree. In this case, we construct a speaker recognition system on the fly consisting of the speakers occurring within the segment (e.g. Patrice followed by George followed Patrice). The system is trained on all the segments that contain only a single speaker. Running the speaker recognition gives us the start time and end time for each speaker within the segment. Combined with speech recognition to generate the start and end times of each word, we now have a segmentation of the words and actors for the entire episode.

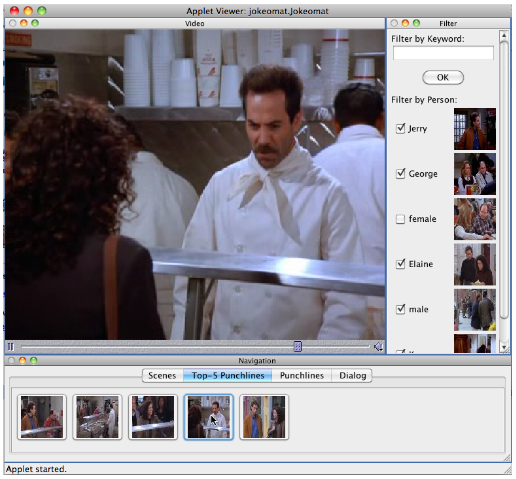
When measured against expert annotation, we obtained a Diarization Error Rate of 24.6%. We also conducted a small user study, asking participants to rate Joke-o-mat HD (using fan-sourced segmentations) vs. Joke-o-mat (using expert-generated segmentations). Most users expressed no preference. Those that did express a preference were almost evenly split between the two.

2.4 Narrative Theme Analysis

The narrative theme analyzer is a rule-based system that transforms the segmentation generated by any of the three methods described above into segments that reflect narrative themes. The rules for the *Seinfeld* themes are as follows: A *dialog element* is a single contiguous speech segment by one speaker. A *punchline* is a dialog act that is followed by pure laughter. Punchlines are prioritized by the length of this laughter segment. The longer the laughter, the more important is the punchline. The *top-5 punchlines* are the 5 punchlines followed by longest laughter. A *scene* is a segment of at least 10 seconds between two music events or a music event and the beginning or end of file.

2.5 Video Browser

We consider the browser (see Figure 3) as a replacement for the typical YouTube video player. The browser shows the video and allows play and pause, as well as seeking to random positions. The navigation panel on the bottom shows iconized frames of the video. The frames are grabbed at 50 % duration of the narrative element that it represents. The navigation panel allows the user to directly jump to the beginning time of either the scene, punchline, top-5 punchline, or dialog element. Also, the current narrative element is highlighted while the show is playing. In order to make navigation more selective, the user can type in a keyword. Only scenes, punchlines, and dialogs that contain the keyword are displayed. The user can also deselect one of the main actors or the male/female supporting actors. In this case, scenes, punchlines, or dialogs that only contains deselected actors are no longer visible in the navigation bar. The actor icons are currently grabbed from the center of their longest dialog segment; we imagine using the localization approach presented in [2] to obtain better results in the future.



☑ Jerry
☑ George
□ female
☑ Elaine
☑ male

Applet started.

Figure 3: The narrative theme video browser.

3. CONCLUSION AND FUTURE WORK

We demonstrated Joke-o-mat HD, a system to segment and browse sitcoms according to narrative themes and compared expert vs. automatic vs. human derived "fan-sourced" segmentations. Future work includes integration of video techniques to improve scene segmentation (e.g. for shows that do not use musical interludes to mark scene transitions), video localization to extract actor icons, and more complete comparison between expert, automatic and fan-sourced methods with respect to the words spoken. We are also interested in task-based methods to evaluate the performance of the various systems (such as finding a specific punchline or summarizing an episode).

4. REFERENCES

[1] G. Friedland and O. Vinyals. Live speaker identification in conversations. In *Proceedings of ACM Multimedia*, pages 1017–1018. ACM, October 2008.

[2] G. Friedland, C. Yeo, and H. Hung. Visual speaker localization aided by acoustic models. In *Proceedings of ACM Multimedia*, pages 195–202. ACM, October 2009.

[3] A. Stolcke, X. Anguera, K. Boakye, Ö. Çetin, A. Janin, M. Magimai-Doss, C. Wooters, and J. Zheng. The SRI-ICSI Spring 2007 Meeting and Lecture Recognition System. Multimodal Technologies for Perception of Humans, Lecture Notes in Computer Science, pages 450–463, 2008.