Automatic retrieval of closed captions for web clips from broadcast TV content

Chris J. Hughes & Mike Armstrong

BRITISH BROADCASTING CORPORATION
Automatic retrieval of closed captions for web clips from broadcast TV content

Chris J. Hughes & Mike Armstrong

Abstract

As broadcasters' web sites become more media rich it would be prohibitively expensive to manually caption all of the videos provided. However, many of these videos have been clipped from broadcast television and would have been captioned at the point of broadcast.

The recent FCC ruling requires all broadcasters to provide closed captions for all 'straight lift' video clips that have been broadcast on television from January 2016. From January 2017 captions will be required for 'Montages' which consist of multiple clips, and the requirement to caption clips from live or near-live television will apply from July 2017.

This paper presents a method of automatically finding a match for a video clip from within a set of off-air television recordings. It then shows how the required set of captions can be autonomously identified, retimed and reformatted for use with IP-delivery. It also shows how captions can be retrieved for each sub-clip within a montage and combined to create a set of captions. Finally it describes how, with a modest amount of human intervention, live captions can be corrected for errors and timing to provide improved captions for video clips presented on the web.

This document is based on one originally published at NAB 2015. Because it was written for an American audience it uses the word "captions" throughout in place of "subtitles".

Additional key words: subtitles
Introduction

In the UK the BBC’s video on demand service, known as iPlayer, provides captioning for a minimum of 98% of the programmes available, but less than 0.1% of the video clips taken from broadcast programmes on the BBC websites are available with closed captions. It would be prohibitively expensive to have all this content captioned manually. However, much of this content has been taken from footage that has been, or will be, broadcast on linear BBC TV channels and all BBC programmes are provided with captions when broadcast.

Over the past two years we have been exploring ways in which we can repurpose broadcast captions to provide captions for web clips. Around the same time groups representing people with hearing loss have started to focus their attention on the need for captioning on web clips [1]. Meanwhile, in the USA, the Federal Communications Commission has published requirements for the captioning of clips from broadcast TV programme from January 2016 [2]. This has provided further incentive to progress this work, the outcome of which is described in this paper.

Overview

Our approach creates a caption file for a web clip by identifying the time at which the clip was broadcast and extracting the captions for this time-segment from the broadcast archive. These captions are then reformatted into a form suitable for reuse along with error correction and retiming. This approach could also be used to provide matching subtitles for any other application where extracts from broadcast programmes are reused, such as making a highlights programme or a programme trailer.

In the case of a video clip on a web page, the speed of the search is improved by using the data in the web page and its context to reduce the number of programmes that need to be searched. Such data includes the creation time and date of the page, text describing the clip or the story it relates to, and in a few cases the web page may contain the programme title and data of broadcast.

The general method is to perform a comparison between the video clip and the contents of the archive/library. The audio signal is preferred in this case because it relates directly to the words in the captions, whereas the same video clip may have been broadcast with a different soundtrack. This work focuses on matching the audio extracted from the web clip to audio stored in a broadcast archive.

Our implementation makes use of a fingerprinting algorithm to represent the audio from the broadcast content and the web clips and stores the results in a database, alongside the matching captions. This speeds up the search of the database for the matching clips whilst providing sufficient accuracy to find the matching captions. However, this search could also use video fingerprints and any other metadata to identify when and where the clip was (or will be) broadcast. A video search would, for example, be more appropriate when looking for data relating to the visual content to accompany a clip, such as with video description.

The BBC web site contains many clips from older programmes where a digital archive of subtitles is available, but the programme archive is on tape or film. In this case it may be possible to use speech to text technology to generate an approximate script for the clip and use language processing to directly locate the matching subtitles, but this would need to be the subject of further research.
3 System Components

Our prototype system builds on existing tools and utilises a number of techniques in order to locate a section of an existing broadcast that matches a video clip presented separately within a web page. Having identified the time of broadcast of the matching video clip, the captions created for broadcast can be retrieved and repurposed to accompany the web video.

3.1 REDUX

BBC Redux is an archive of broadcast content developed as a research tool and test-bed by BBC R&D. The system has been recording the BBC's DVB streams off air continuously since mid-2007 [3][4]. The system separates the streams into the individual TV channels, and then into individual TV and radio programmes using the broadcast schedule data. The recorded programmes can either be accessed manually via a web interface or through a software API. Each programme can be identified by name, broadcast time and keywords from the programme description. The audio and video can be extracted from the stored transport stream along with the broadcast captions, video description and other data.

3.2 Heuristics

Each web page that contains a video clip will also contain text and metadata that can be used to speed up the search of the broadcast archive. This data is used to create a search path through the archive, by weighting each entry based on a number of criteria, for example: the time the webpage was created, most likely broadcast channels and keywords. This enables the search to be performed as an interactive process, generally returning results within a few seconds. Without this optimisation search times could be significantly longer and in the worst case it could mean searching the entire archive.

3.3 Fingerprinting

Captions primarily represent the audio component of a programme, so we have chosen to perform the search by comparing the audio in a clip with the audio in the television programmes. To reduce the amount of data required to represent both the programmes and the clips we have chosen to use a form of audio fingerprinting which represents the audio with a time resolution of around one second. This significantly reduces search time while maintaining a high level of accuracy and sufficient time resolution for caption alignment.

Chromaprint [5] was used as our primary fingerprinting tool. This is available as open source under a LGPL2.1+ licence. This algorithm produces a unique fingerprint for a 30-minute programme in about 200kb. This provides a reduction of around 2% over storing the original audio file, even after it has been compressed into the MP3 format. The fingerprints are generated as a series of integer values using a python [6] script.

Several studies provide comprehensive reviews of alternative fingerprinting tools for both audio and video which could be used to explore alternative algorithms [7], but that is beyond the scope of this work.
3.4 Edited Clips

A match for the whole clip will fail if it is a montage made up of several pieces from separate parts of one or more programmes. To overcome this limitation we repeat the search but this time initially search only for the first few seconds. If we find a match, in the archive, the fingerprint length is gradually increased and compared to the matching programme until the fingerprint no longer matches. This is then interpreted as the boundary at which the video was edited. Captions are returned for the first section and the process is repeated for the next sub clip, starting from the edit, until each section of the clip has been identified as illustrated in Figure 1.

![Web Clip: Sub Clip 0 Sub Clip 1 Sub Clip 2](image)

Figure 1 – Captions can be identified within a montage by identifying a number of sub clips.

3.5 Caption Processing

Once a match has been identified a request is made to the archive for the captions for the programme from a point a few seconds before the start of the match to a point a few seconds after the end of the clip. This ensures that we have not missed the start and end of the required subtitles and we can intelligently trim the captions based on the sentence structure. In the case of captions that are created live there is the additional problem of the delay in the live subtitling process, where live captions are usually several seconds behind the speech. In the case of live captions in the UK lines of text are generally repeated as they are scrolled upwards, so it is also necessary to remove any repeats and format the captions into blocks.

3.6 Web Based Editor

Live broadcast captions often contain errors due to the time constraints on their production. In order to provide a means of correcting these errors and a degree of editorial control our prototype system also includes a basic web-based editor. This enables the manual adjustment of the caption text and simple retiming. This process is assisted by the system, which automatically conforms the caption blocks according to appropriate guidelines [8].
4 Implementation

Our prototype implementation is built using a server that maintains and updates the database of fingerprints, captions and related data. It also provides services for searching programmes and returning caption data. The server polls the archive to identify when new programmes have been added following broadcast. When a new programme appears the matching audio is cached and a fingerprint is generated for the entire programme. This fingerprint is then written to the database, along with the matching captions. Other data such as channel and programme name are also added to the database.

The BBC archive, Redux, is built around the idea of providing a just-in-time service and only extracts the audio and captions from the transport stream on demand. This means that there can be a further delay between the end of the programme and the file becoming available for fingerprinting. Our prototype attempts to ameliorate this issue by parallelising the download of the captions and audio enabling multiple programmes to be downloaded simultaneously while others are fingerprinted. A future implementation of our technique could streamline the process by additionally taking a feed directly off-air, and producing fingerprints in near real-time directly from the broadcast.

The search mechanism is invoked by using a URL which points to a web page containing a video clip that requires captions. Once the search starts the server visits the URL, downloads the video clip from the page, extracts the audio and generates a fingerprint representation for the clip. It also extracts data from the web page including the creation time and date for the page, as shown in Figure 2. This data is then used to generate the heuristic search path through the archive, by weighting each of the database entries based on time, date, channel and keywords found.

The search algorithm then performs a brute force search through the database, using cross correlation [9] between the clip fingerprint and each programme entry in the database. The cross correlation algorithm performs a sliding difference comparison between the sub fingerprint a within each database entry fingerprint b as shown in (1).

\[
s = \sum_{i=n}^{(n+f)} |a_i - b_i| = |a_1 - b_1| + |a_2 - b_2| + \ldots + |a(n+f) - b(n+f)| \quad (1)
\]

Figure 2 – Example data derived from a web page stored in JSON format (note the error "A women" in the description was present in the original web page metadata, not displayed text)
A sliding window is used to cross correlate the clip fingerprint against each of the programmes sequentially along the ordered search path and the result provides a confidence value for the match.

Two thresholds are used to rate the success of the match. If the correlation meets the first threshold it is assumed to be a highly likely match and returns an immediate result. If the first threshold is not reached, each correlation that meets the second threshold it is marked as a possible match and once the search has completed, the best match is returned. If no match is found the search can be manually re-invoked to perform the lengthier search for edited clips.

![Excerpt from Fingerprint representation of Broadcast Programme]

![Fingerprint representation of Search Clip]

![Search Clip Correlated to Broadcast Programme]

Figure 3 – The search clip fingerprint (red) is matched to the fingerprint of the broadcast programme that it was taken from (blue).

A successful search returns the start and end times for either the whole clip or each of the sub clips identified. These timings can then be used to retrieve the captions as broadcast.

Throughout the search process caching is employed to speed up requests. A database contains all downloaded video files and their related assets. Resulting caption files are also stored and directly provided if the clip has previously been successfully identified. Other assets cached include an audio waveform graphic for each audio file, which is then used in the timeline of the caption editor.

The search process is also parallelized to make use of the multiple cores available on the server. The search path is partitioned and distributed to a worker for each core, which then compete to find the highest confidence value.
5 Caption Processing

Once the captions have been retrieved the system performs text processing to remove the repeated lines in live captioning and natural language processing is used to identify the most likely start and end of the captions. Also, where the caption author provided a correction during a live broadcast (usually indicated by "--" in the UK) the correction is applied to the original text. Colours are used in the UK to identify different speakers and these are remapped to ensure that the clip follows the conventional colour order, as shown in Figure 4.

<table>
<thead>
<tr>
<th>Speaker</th>
<th>Broadcast Colour</th>
<th>Reassigned Colour</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>#ffff00</td>
<td>#ececcececececece</td>
</tr>
<tr>
<td>2</td>
<td>#ececcececececece</td>
<td>#ffff00</td>
</tr>
<tr>
<td>3</td>
<td>#00fffd</td>
<td>#00fffd</td>
</tr>
<tr>
<td>4</td>
<td>#00ff00</td>
<td>#00ff00</td>
</tr>
</tbody>
</table>

Figure 4 - Colour lookup table generated to reassign caption colours into the traditional order.

The captions are then formatted into two-line blocks. This is guided by a maximum line-length rule. By default this is set at 37 characters as constrained by the original Teletext line length limitations [10], although can be overridden by the user. The process works by appending each word to a line until the next word takes the character count over the limit, at which point a line break is inserted and a new line is started. Additional rules are used to avoid leaving orphaned words, by forcing a new line if a punctuation mark is added over half way through the line. Breaks are always inserted when there is a change of speaker. The lines are then assembled into two line pairs to complete the block as shown in Figure 5.

Figure 5 - The captions are re-blocked based on maximum character per line limit.
Finally the captions are retimed. As the captions have been clipped from the file for an entire programme, each of the timings are relative to the start of the programme and not the start of the clip. Therefore the time of the start of the clip is subtracted from the original timings to provide a rough alignment with the clip. In the case of live captions these timings will be inaccurate because of the delay, so further processing is required. This is done by making the assumption that short clips are always cut quite tightly to the dialog. The timing is then improved by adjusting the timing of the first caption to start at the beginning of the clip and moving the last caption so that it ends at the end of the clip. Each caption in between is then adjusted to fit, maintaining the relative timings from the broadcast.

At this point the large majority of the clips can be viewed with automatically retrieved and retimed captions and they can provide a satisfactory viewing experience, see Figure 6. In the case where the original captions were from a live broadcast, the results are generally better than the original as a result of the improved timing and any corrections. For someone who requires captions to view the video clip this is a vast improvement over having no captions at all. However, most errors will remain and a manual process is needed to remove them.

Figure 6 – Although the automatically returned captions contain some errors carried through from the live broadcast, they would still provide a satisfactory viewing experience.
6 Caption Editor and Enhanced Viewer

Because the automatic system cannot be completely accurate and live captions will still contain errors, which cannot be identified automatically, a final viewing and editing stage has been built. The editor allows the captions to be reviewed and provides an interface for the manual correction of the remaining errors. For example in Figure 6, in the first caption the word ‘High’ was incorrectly captioned as ‘I’ and in the fourth caption the word ‘teams’ has been incorrectly identified as ‘teens’. Figures 7 & 8 show the captions overlaid on the video, before and after the correction.

Figure 7 – A typical phonetic spelling error – the word ‘High’ was incorrectly captioned with the word ‘I’.

Figure 8 - The captions are easily corrected and improved further with a small amount of editorial input.
An enhanced viewer was also built to further test the framework and demonstrate how the output was generated. This stage has been built as a web-based tool. The viewer provides the user with a video player with caption playback, an annotated timeline and basic navigation controls to enable the video to be played back and a qualitative assessment made of the caption alignment as shown in Figure 9. It also allows captions to be edited and manually retimed quickly as required in order to maintain an editorial control.

Figure 9 – The editor allows for the captions to be manually corrected and realigned with the audio waveform.

It includes options, such as a one-click manual retiming and the ability to import text from other sources, such as transcripts and original scripts. It also provides warnings to indicate whether the current captions meet guidelines, such as the reading rate for each caption.

This tool has been designed for use by journalists and producers who produce web clips rather than as a full caption editor so it also attempts to provide basic metrics, such as warning if the number of words per minute exceeds the guidelines [8].
7 System Performance

Our initial work has been focused on content available on the BBC News Website (www.bbc.co.uk/news). We have found that the heuristic algorithm is sufficiently effective to narrow down the search to the point where the clip is generally found within the first 4 programmes in the list.

On average it takes 3.65 steps through the search path with a result returned in about 0.6 seconds. The maximum number of programmes searched, for a positive result, was 17 returning in under 3 seconds. The default maximum number of programmes it will search is set to 100 therefore if no result is found, the search returns within around 17 seconds.

![Figure 10 – Time taken to search different length programmes.](image)

Taking a snapshot of the BBC News web site at one point in time, the site was found to contain 346 video clips. Of these around 40% were broadcast in the same form, i.e. the clip had not been edited and therefore could be found automatically on a first search. When the search was extended to look for edited clips it identified around 20% were built up from separate sub clips, extending the positive result to around 60%. So no match was found for around 40% of the web site. The most common causes for these failures were identified, as either the clip had not been broadcast or the clip was too heavily edited for each section to be identified. Clips need to be at least 5 seconds long to produce a unique fingerprint, so this is the shortest sub-clip that can be matched. In a few cases the clip contained no audio and therefore would not need subtitling.

Occasionally clips are published on the web site prior to broadcast. This is most common with breaking news stories where the clips are published in a web page before the opportunity arises to insert them into a television broadcast. Due to the time pressures on publishing the story, the clip will be used unedited on both web and on TV. This means that some clips may be found by repeating a failed search after a suitable interval. However, is common for such clips to then be edited at some later time and the webpage updated with a new version of the video clip. At this point the captions would no longer match and a new search would need to be performed or the captions updated manually.

We also performed an analysis of the results across the different sub-sections of the BBC News website and found that different sections typically have different editorial requirements, resulting in
differing levels of success, as shown in Figure 11. The highest success rate was found with ‘Entertainment’, ‘Health’ and ‘Science’ where clips were generally taken directly from broadcast. The ‘Technology’ section provided the lowest success rate. This section is heavily curated and contains video clips that often have a large number of edits along with video content that has been made specifically for the web and never broadcast.

Figure 11 – Due to different editorial styles the success rate varies across different parts of the BBC news website.

8 Further Applications

In some cases where it is not possible to find a match, alternative data sources may exist that allow the creation of a set of captions. For example the BBC iWonder Website (www.bbc.co.uk/iwonder) contains a high proportion of video clips which are made specifically for the website, not for broadcast. However, these video clips are published with complete transcripts. We have been able to successfully demonstrate an automatic conversion to captions using our prototype tools. Whilst this technique produces usable results for a short video clip it does not match the standards expected of television caption services.

The approach used to retrieve caption data could also be adapted to retrieve other forms of broadcast meta-data related to a clip. Where the programme contains video description it would be possible to retrieve the video description to accompany a clip on the web. As video description relates to the video content it would be more appropriate to match the clip against a video fingerprint.

This technique could also be used to retrieve other data stored alongside the audio and video content, such as timing triggers for an interactive app or a list of shot changes or script information describing the scene or the characters or actors within the scene.
9 Conclusions

We have built a prototype system, which successfully demonstrates the practicability of automatically providing captions for web clips that have previously been captioned for broadcast. This system functions entirely independently of the television production and broadcast chain using an off-air archive. In the case where the broadcast captions were of high quality we have shown that this process provides accessibility with little need for human intervention. Where the captions were created live to accompany a broadcast and were affected by delay and errors the system can provide improved captions, with minimal editorial intervention.

Our process would be significantly more cost effective than creating the captions from scratch. Only in the case of the remaining clips, which our process cannot find, would new captions need to be made. If the clip has not been broadcast then under the new FCC rules the captions would be optional. Only in the cases where the clips have been heavily edited from broadcast material would provision of new captions would be mandatory.

Our future work will extend to look at the provision of captions for other BBC websites and working towards a practicable implementation of these techniques that can be integrated with the BBC website publication systems.

10 Acknowledgements

This work has been undertaken at BBC Research and Development with encouragement and support from Gareth Ford-Williams, Head of Accessibility for BBC Future Media.

The methodology described in this paper is has been filled under UK patent number-P/72151.GB01 'Subtitling Web Video Clips'.

11 References