A free and open-source tool that reads movie subtitles aloud

Peter Ljunglöf
Computer Science and Engineering
University of Gothenburg
Gothenburg, Sweden
peter.ljunglof@gu.se

Sandra Derbring and **Maria Olsson**
DART: Centre for AAC and AT
Gothenburg, Sweden
sandra.derbring@vgregion.se
maria.in.olsson@vgregion.se

Abstract

We present a simple tool that enables the computer to read subtitles of movies and TV shows aloud. The tool extracts information from subtitle files, which can be freely downloaded from the Internet, and reads the text aloud through a speech synthesizer. There are three versions of the tool, one for Windows and Linux, another for Mac OS X, and the third is a browser-based HTML5 prototype. The tools are freely available and open-source.

The target audience is people who have trouble reading subtitles while watching a movie, including elderly, people with visual impairments, people with reading difficulties and people who wants to learn a second language. The application is currently being evaluated together with user from these groups.

1 Background

1.1 Why read subtitles aloud?

Spoken subtitles could be a solution if, due to sight disorder or poor reading skills, a person is unable to read subtitles and the language spoken in the movie is unknown, or not known well enough.

Swedish Association of the Visually Impaired[1] has around 12,000 members but there are most likely many more people with poor eyesight. The number of people with reading disabilities is unknown, but according to the Swedish dyslexia association "Dyslexiföreningen"[2] between 5 and 8 percent of the population have significant difficulties to read and write. A survey by OECD (Organisation for Economic Co-operation and Development) in 1996 showed that "8 per cent of the adult population [in Sweden] encounters a severe literacy deficit in everyday life and at work" (OECD, 2000, p. xiii). For other countries, the problems were even bigger: "In 14 out of 20 countries, at least 15 per cent of all adults have literacy skills at only the most rudimentary level" (OECD, 2000, p. xiii).

To hear the subtitles along with the original audio track of the movie may not suit everyone, but making these movies and TV shows accessible could bring a huge value for people who would use it.

1.2 Related work

The idea of automatic reading of movie and TV subtitles is not new. It is implemented in regular public service TV broadcasts in at least Sweden and the Netherlands, and probably also in more countries. In 2002, the Dutch national broadcasting company NOS started regular broadcasts of automatic subtitles reading (Verboom et al., 2002), and Sweden's public service TV company SVT followed in 2005 (A-focus, 2010, p. 20). In both these cases, the speech signal is transmitted through a second channel, which means that the user needs two digital boxes. Naturally, this solution only works for the programs that the company itself is broadcasting.

Other projects have been trying to use OCR (optical character recognition) to interpret the subtitles on the TV or computer screen. In 2002, a project by the Swedish Association of the Visually Impaired developed a prototype that used OCR to Interpret subtitles, which then were spoken aloud using TTS (Eliasson, 2005, pp. 63–64). The project estimated that a mass-produced product would cost around 2500€, which they concluded would be too much

[1] Synskadades riksförbund, http://www.srfriks.org/
[2] Dyslexiföreningen, http://dyslexiforeningen.se/

NAACL-HLT 2012 Workshop on Speech and Language Processing for Assistive Technologies (SLPAT), pages 1–4,
Montréal, Canada, June 7–8, 2012.

for ordinary users. In 2007, a similar Danish project described a tool that reads the composite video signal, performs OCR on the subtitles and then speaks them using TTS (Nielson and Bothe, 2007). They also developed a specialised OCR algorithm for subtitle detection (Jønsson and Bothe, 2007). However, both systems have remained prototypes and have not been released as publicly available tools.

A similar Czech project has investigated how to minimise speech overlap and how to get better synchronisation by using techniques such as time compression and text simplification (Hanzlíček et al., 2008; Matoušek et al., 2010). Their evaluation is purely technical, where they count the number of overlapped subtitles and the number of subtitles that require different compression factors, but they have not evaluated their prototype system on actual users.

Finally, there is an ongoing Swedish project by the Swedish dyslexia association "Dyslexiförbundet FMLS" where they aim to make cinemas more accessible by transmitting spoken subtitles via Wi-Fi which the users can listen to via their own mobile phone.

1.3 Issues with existing solutions

Currently there are two kinds of spoken subtitles systems, and both of them have different problems:

- TV broadcasting systems that transmit the spoken subtitles in a separate audio stream. It is an important addition to the TV infrastructure, but it is by nature closed to one media channel and cannot be used for users who want to watch movies or TV shows on their computer or from the Internet.

- Systems that use OCR to interpret movie subtitles have a great potential, but they are currently no publicly available systems. There are still some technological problems left to be solved until OCR based systems can be released to the public.

None of the existing systems are freely available, let alone open-source products. Furthermore, we have not found any studies that evaluate these systems on real users, to find out how useful they are in practice.

The systems we describe in this paper are all freely available and open-source. They are focused on personal computer use, not TV or cinemas, and are meant to be usable and easily installable to those with basic computer skills.

2 Implementation

The idea behind all our implementations is very simple. The program reads the subtitles into an internal database. When the movie starts playing, the program communicates with the movie to get the current time position, and calls a speech synthesiser when it is time to show the next subtitle. The program does not include a speech synthesiser, but assumes that it is already installed on the computer. Alternatively, the program can call an online web service-based TTS.

We have developed three systems which work in different ways and on different operating systems. Some of them are still in prototype/demo state, whereas others are almost finished products. All systems are free and open-source and can be downloaded from the project website:

http://code.google.com/p/subtts

2.1 Windows/Linux media player

The Windows/Linux client has been developed by the company STTS.[3] It is implemented in Python and the wxWidgets GUI toolkit.[4] The video playback interface uses a Python backend that comes with the VLC Media Player.[5] This means that the client can play all media formats that the VLC player can handle, including DVD movies.

2.2 Mac OS X menulet

The Mac OS X client uses the AppleScript Event model to communicate with the active media player. The program is developed in Objective-C and resides in the menu bar as a global "menulet"[6].

When the user starts watching a movie, the menulet repeatedly queries the media player for the current time, and calls the speech synthesiser whenever a new subtitle is about to be shown. The menulet currently supports the following media players: VLC, QuickTime Player (versions 7 and X), and Apple DVD Player.

[3] Södermalms Talteknologiservice, http://stts.se/
[4] wxWidgets, http://wxwidgets.org/
[5] VLC, http://videolan.org/
[6] http://en.wikipedia.org/wiki/Menulet

2.3 Browser-based HTML5 media player

We have also developed a prototype browser-based media player written in Javascript, that uses HTML5 video and audio elements to support spoken subtitles. This has the potential to be very useful, but is currently limited since current browsers do not support HTML5 video and audio in full.

We estimate that, in a few years time, the main browsers will support all HTML5 features,as well as offline TTS. Then this kind of HTML5 media player could have a big impact on movie and TV accessibility.

2.4 Subtitle files

The system does not extract the subtitles from the movie file or the DVD. Instead the user has to provide it with a text file with the movie subtitles. Subtitles are available from several sites on the Internet,[7] both in the original language and in translations into different other languages.

The subtitle format that we support is SRT, which is the de-facto standard for movie subtitiles and a very simple text format. Each subtitle is in a separate paragraph on the following form:

```
26
00:03:05,083 --> 00:03:09,417
You, I mean we,
we could easily die out here.
```

The above example means that the 26th subtitle consists of two lines of text, and should be displayed 3 minutes 5.083 seconds into the movie and disappear 4.334 seconds later.

Both the Windows and the Mac OS X clients can show DVD movies, but they cannot use the subtitles that are provided with the movie. DVD subtitles are pre-rendered into separate video tracks. To access them we would have to use OCR which was not in the scope of this project.

One serious drawback with existing subtitles is that they do not store meta-information about the speaker. Useful meta-information would be gender, age and dialect of the speaker, or even a unique identifier for each person in the movie. With this information the system could use different TTS voices for different characters.

[7] E.g., http://opensubtitles.org/ and http://undertexter.se/.

2.5 Speech synthesis

We are only using existing speech synthesisers, which means that the user either has to have a TTS voice installed on his/her computer, or constant access to the Internet since the system can call existing online TTS engines. The only problem with online TTS systems is that almost all of them are for demonstration purposes only and therefore cannot be used in day-to-day work. We have been using an online Swedish open-source voice being developed by the company STTS[8] using the OpenMary TTS platform (Schröder and Trouvain, 2003).

Here is the current status of speech synthesis for our different systems:

- The Windows client can use any SAPI voice installed on the system. It can also use an online voice, as an alternative.
- The Mac OS X client can use any voice installed on the system. The latest version of OS X (10.7) includes high-quality voices for 22 different languages, so there is no need for online voices on this platform.
- The HTML5 browser client cannot use system-installed voices, since that functionality is not included in HTML5. There is a current W3C draft proposal for how to use TTS from within HTML (Bringert, 2010), but it is not decided upon and no browsers support this yet. Until TTS becomes a HTML standard we have to rely on online voices, which unfortunately is a scarce resource.

3 Discussion

3.1 Social and pedagogical advantages

People with visually impairments and/or reading difficulties often use text-to-speech to cope with school work, and to keep up with society. Spoken subtitles further increase the accessibility of foreign movies and TV shows for these people.

Hopefully, spoken subtitles can help improve the reading skills for people with reading difficulties. The theory is that listening to the spoken subtitles at the same time as reading the text may benefit the reading process, but this has yet to be tested.

[8] Södermalms Talteknologiservice, http://stts.se/

3.2 Evaluation

We are currently, during spring 2012, evaluating the applications together with different users in the target groups. Initially we will only be evaluating user satisfaction and whether this approach could be an accepted solution to the need of text interpretation during movie playback.

If this initial evaluation is positive, we are very interested in continuing by evaluating specific factors that might or might not improve user satisfaction. Such factors could be: using different TTS voices, using different speech rates, reducing speech overlap, having the speech coming from another direction, lowering the movie volume while speaking, using advanced audio techniques for filtering away movie speech, etc.

Another interesting evaluation would be to encode speaker meta-information into movie subtitles, and test how different TTS voices for different characters can improve the user's satisfaction and comprehension.

3.3 Future work

To further ease the user friendliness and the availability, it would be desirable to have the functionality built into an existing media player, such as the open-source and cross-platform VLC Media Player.[9] If more users request this functionality, the developers will have to catch on and include it into new releases.

According to (Hanzlíček et al., 2008), 44 percent of the Czech subtitles had overlaps when spoken with TTS. Even though we have no figures for Swedish, some overlap is to be expected also here, which is an issue that should be addressed. One possible simple solution is to modify the speech rate.

An important factor for the experience of the speech synthesizer together with a video playback would be the settings of the audio channels. Hypothetically, a listener would want to keep both the original background cues, like music, and the original voices. However, these sounds must not interfere with the speech synthesizer that is the source of information for the listener. Balancing these two criteria to get the optimized result is of great interest.

If the program would be used for language learning, or to help slow readers to comprehend, the feature of highlighting the word that is spoken could be a very useful additional feature.

[9] VLC Media Player, http://www.videolan.org/vlc/

Acknowledgements

The SubTTS project is funded by the Swedish Post and Telecom Authority (PTS). We are grateful to four anonymous referees for their comments.

References

A-focus. 2010. *Utredning avseende TV-tillgänglighet för personer med funktionsnedsättning*. Myndigheten för radio och TV, Stockholm, Sweden.

Björn Bringert. 2010. HTML text to speech (TTS) API specification. W3c editor's draft, W3C.

Folke Eliasson. 2005. *IT i praktiken – slutrapport*. Hjälpmedelsinstitutet, Sweden.

Zdeněk Hanzlíček, Jindřich Matoušek, and Daniel Tihelka. 2008. Towards automatic audio track generation for Czech TV broadcasting: Initial experiments with subtitles-to-speech synthesis. In *ICSP '08, 9th International Conference on Signal Processing*, Beijing, China.

Morten Jønsson and Hans Heinrich Bothe. 2007. OCR-algorithm for detection of subtitles in television and cinema. In *CVHI'07, 5th Conference and Workshop on Assistive Technology for People with Vision and Hearing Impairments*, Granada, Spain.

Jindřich Matoušek, Zdeněk Hanzlíček, Daniel Tihelka, and Martin Méner. 2010. Automatic dubbing of TV programmes for the hearing impaired. In *10th IEEE International Conference on Signal Processing*, Beijing, China.

Simon Nielson and Hans Heinrich Bothe. 2007. SubPal: A device for reading aloud subtitles from television and cinema. In *CVHI'07, 5th Conference and Workshop on Assistive Technology for People with Vision and Hearing Impairments*, Granada, Spain.

OECD. 2000. *Literacy in the Information Age: Final Report of the International Adult Literacy Survey*. OECD Publications, Paris.

Marc Schröder and Jürgen Trouvain. 2003. The German text-to-speech synthesis system MARY: A tool for research, development and teaching. *International Journal of Speech Technology*, 6:365–377.

Maarten Verboom, David Crombie, Evelien Dijk, and Mildred Theunisz. 2002. Spoken subtitles: Making subtitled TV programmes accessible. In *ICCHP'02, 8th International Conference on Computers Helping People with Special Needs*, Linz, Austria.