

## BACKGROUND

Anyone who ever downloaded **subtitle files** from the Internet has faced problems synchronizing them with the associated media files.

Even with the efforts of communities on reviewing user-contributed subtitles and with mechanisms in movie players to automate the discovery of subtitles for a given media, users still face **lip synchronization issues**.

In this work we conduct a study on several subtitle files associated with popular movies and TV series and analyze their differences.

Based on that, we propose a two-phase **subtitle synchronization method** that annotates subtitles with **audio fingerprints** which serve as synchronization anchors.

## THE PROBLEM

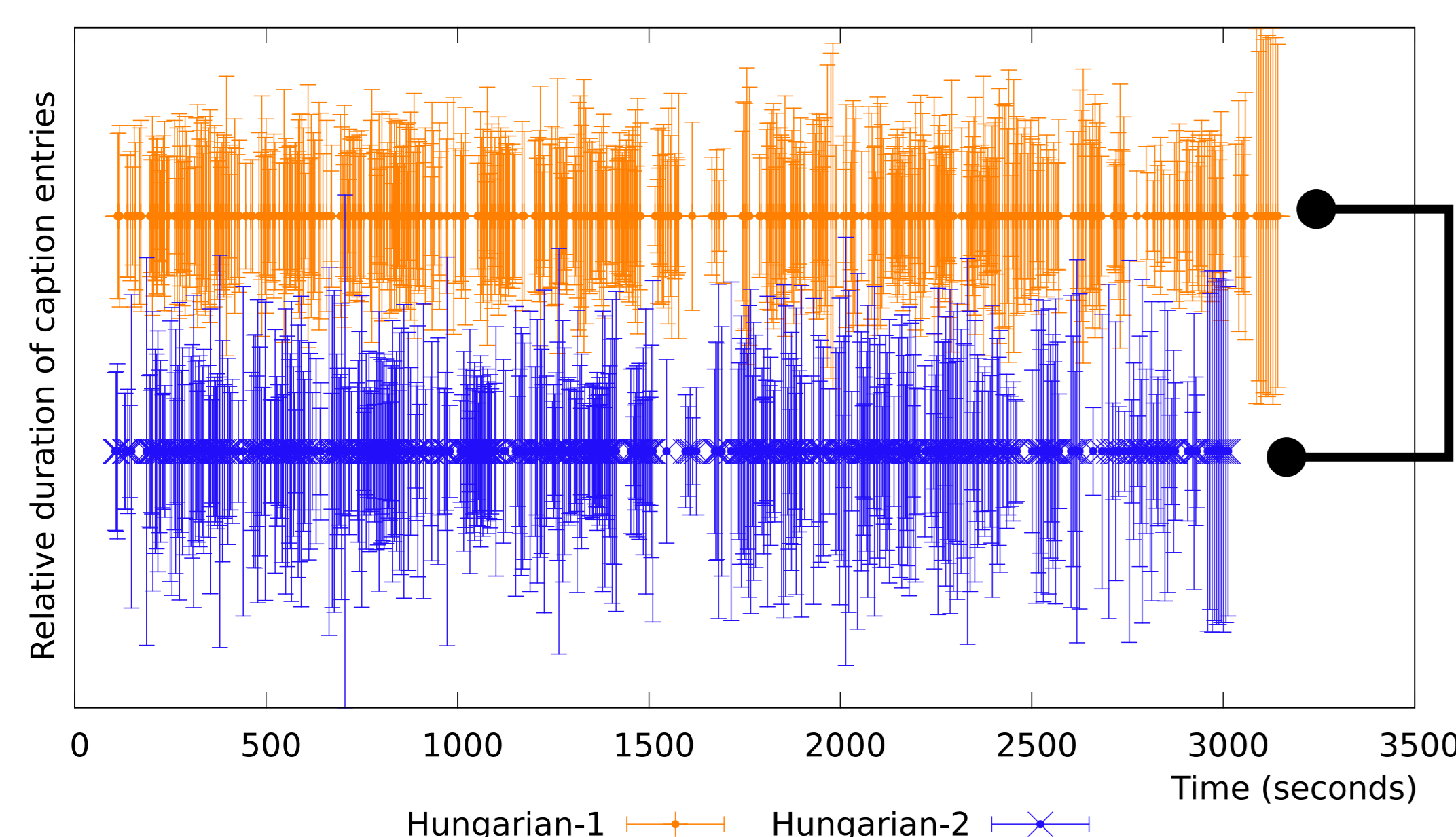
01. ...  
02. 615  
03. 00:50:02,280 --> 00:50:06,046  
04. I wish I had enough poison  
05. for the whole pack of you.  
06.  
07. 616  
08. 00:50:06,120 --> 00:50:10,489  
09. I would gladly give my life  
10. to watch you all swallow it.  
11.  
12. 617  
13. 00:50:10,693 --> 00:50:12,711  
14. [crowd shouting]  
15. ...

FILE A

01. ...  
02. 729  
03. 00:48:25,351 --> 00:48:28,937  
04. I wish I had enough poison  
05. for the whole pack of you.  
06.  
07. 730  
08. 00:48:28,971 --> 00:48:31,139  
09. I would gladly give my life  
10.  
11. 731  
12. 00:48:30,141 --> 00:48:33,475  
13. to watch you all swallow it.  
14. ...

FILE B

Two distinct captioning files, **different synchronization offset!**



Sometimes, videos are encoded with **different frame rates**.

Temporal differences between caption entries of two such videos begin small but increase over time

## OUR SOLUTION: SUBTITLE CREATION

In the subtitle authoring software, one or more subtitle entries are elected as synchronization anchors

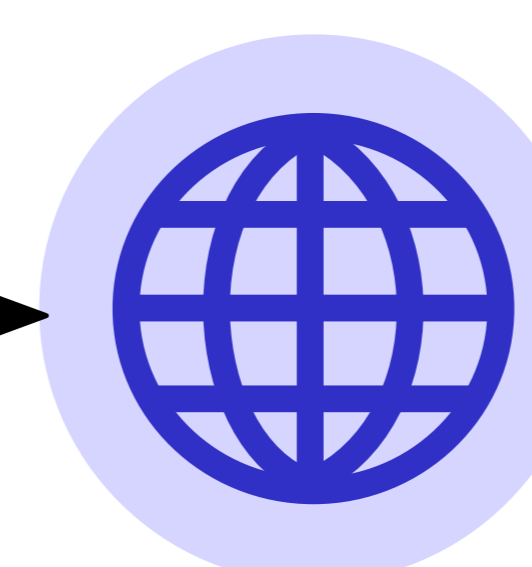
Num	Anchor	Start	End	Text
72	No	05:24	05:26	You're the Lord Commander now.
73	Yes	05:26	05:27	Command!
74	No	05:27	05:29	Let others do the fighting.
75	No	05:29	05:31	When was last time
76	No	05:31	05:33	Father used a sword?
77	No	05:33	05:34	I'm not Father.
78	No	05:34	05:35	I'm the Kingslayer.

We compute the audio fingerprint at the starting time of the elected anchors and store them as meta-data in the generated subtitle file

To overcome the lack of support for including meta-data in the SRT format, we embed audio fingerprints in caption entries with zero duration

```
301. 72
302. 00:05:24,000 --> 00:05:26,000
303. You're the Lord Commander now.
304.
305. 73
306. 00:05:26,000 --> 00:05:26,000
307. @fingerprint@ -177,-168,-167,...,934,934,904
308.
309. 74
310. 00:05:26,000 --> 00:05:27,000
311. Command!
```

The resulting subtitle file is posted to a repository where other users can obtain it



## OUR SOLUTION: PLAYBACK TIME

Another person obtains the annotated subtitle file and instructs the media player to play it along with a media file.

The media player seeks the media file stream to the time offsets of the **first and last anchors** and computes the audio fingerprints at these locations. The medias are in sync if the reference and computed fingerprints match.

If they don't match, then the media player **seeks to the beginning of the media file** and computes its audio fingerprints until there is a match with the signature of the first anchor. The **difference** between the media player's time offset and the anchor's is then **propagated** to all subsequent caption entries.

A comparison with the last anchor's fingerprint is then performed. If the calculated fingerprint at that anchor's offset match with the reference, the medias are in sync.

If they do not match, then the media player computes the audio fingerprints starting at the offset of the first anchor and keeps propagating differences until the last anchor is processed. Finally, the player uses the updated caption timestamps to synchronize the playback.

## PROTOTYPE

The method proposed in this paper has been prototyped with the **VLC media player** and with **Chromaprint**, a library that implements a custom algorithm for extracting fingerprints from audio sources.

Our prototype includes support for the SRT format, although other formats can be easily extended to support our proposal.

## CONCLUSIONS

Through a qualitative analysis of publicly available subtitles **we identified sources of synchronization problems** between captioned files and their corresponding media.

**We developed a method to dynamically synchronize subtitles** based on **audio fingerprints**.

We proposed an **extension to the subtitle file formats** currently in use so they can benefit from our dynamic synchronization method.