Martín, A., González-Carrasco, I., Rodriguez-Fernandez, V. et al. Deep-Sync: A novel deep learning-based tool for semantic-aware subtitling synchronisation. *Neural Comput & Applic* (2021).

DOI: 10.1007/s00521-021-05751-y

Deep-Sync: A novel deep learning-based tool for semantic-aware subtitling synchronisation

Alejandro Martín[1] · Israel González-Carrasco[2] · Victor Rodriguez-Fernandez[1] · Mónica Souto-Rico[3] · David Camacho[1] · Belén Ruiz-Mezcua[2]

Abstract

Subtitles are a key element to make any media content accessible for people who suffer from hearing impairment and for elderly people, but also useful when watching TV in a noisy environment or learning new languages. Most of the time, subtitles are generated manually in advance, building a verbatim and synchronised transcription of the audio. However, in TV live broadcasts, captions are created in real time by a re-speaker with the help of a voice recognition software, which inevitability leads to delays and lack of synchronisation. In this paper, we present Deep-Sync, a tool for the alignment of subtitles with the audio-visual content. The architecture integrates a deep language representation model and a real-time voice recognition software to build a semantic-aware alignment tool that successfully aligns most of the subtitles even when there is no direct correspondence between the re-speaker and the audio content. In order to avoid any kind of censorship, Deep-Sync can be deployed directly on users' TVs causing a small delay to perform the alignment, but avoiding to delay the signal at the broadcaster station. Deep-Sync was compared with other subtitles alignment tool, showing that our proposal is able to improve the synchronisation in all tested cases.

Keywords TV Broadcasting · Synchronisation · Language model · Deep neural networks · Machine learning

1 Introduction

Subtitles are one of the most important elements in making TV as accessible as possible for people with hearing impairment, elderly people and other end-users that could need from this kind of service. For those with hearing impairment disability, reading subtitles in the screen is the procedure for receiving at least most of the audio content being transmitted. In films, television series, videos or pre-recorded content, the process to generate these captions is simple, although very expensive, both in time and cost. Before broadcasting, a subtitling expert writes the subtitles in a manner that ensures the synchronisation with the audio-visual content. However, this process becomes problematic when dealing with live broadcasts. In these cases, a person called re-speaker reinterprets the content heard, introducing different changes due to lack of time, while a voice recognition software generates the corresponding text. This is a useful technique to generate accurate and of high-quality subtitles in real time, but involves two serious problems: on the one hand, the re-

✉ Alejandro Martín
alejandro.martin@upm.es

Israel González-Carrasco
igcarras@inf.uc3m.es

Victor Rodriguez-Fernandez
victor.rodriguezf@uam.es

Mónica Souto-Rico
msouto@cesya.es

David Camacho
david.camacho@upm.es

Belén Ruiz-Mezcua
bruiz@inf.uc3m.es

[1] Departamento de Sistemas Informáticos, Universidad Politécnica de Madrid, Madrid, Spain

[2] Computer Science Department, Universidad Carlos III de Madrid, Av. Universidad, 20, 28915 Leganés, Madrid, Spain

[3] Culture Area, Spanish Centre for Subtitling and Audio Description (CESyA), Spanish, Spain

speaking process introduces a delay, causing the subtitles to not being synchronised with the audio-visual content and, on the other hand, the re-speaker could generate new subtitles that simply does not match with the audio. This includes summarising, paraphrasing, omission of words or even providing a completely different description but semantically similar to the original content. This seriously impoverishes audience experience.

A solution proposed in the state-of-the-art literature to the above highlighted problem is to establish a delay in the broadcasting. This would allow to synchronise the subtitle before being emitted, but involves an ethical issue. Since the content is not immediately sent and it can be modified, this can be considered as censorship [27]. Besides, building a perfect alignment between subtitles and the audio-visual content is a hard task and would require a long delay.

In this paper, we propose Deep-Sync,[1] a tool for the automatic alignment of subtitles and transcriptions that solves the problems listed above. This tool readjusts the presentation and deletion time of every subtitle in order to achieve a perfect synchronisation with the audio-visual content. On the one hand, Deep-Sync can be run as a complement on the user's TV, so it is not necessary to delay the broadcasting at the origin, avoiding any kind of censorship. On the other hand, Deep-Sync integrates a novel *semantic-aware alignment method* that allows to improve the synchronisation process and to achieve high accuracy. Besides, Deep-Sync can operate in various scenarios, dealing with pre-recorded, live-scripted or live-improvised, or in a mixed scenario with a combination of them in the same broadcasting.

Deep-Sync takes advantage of an automatic speech recognition (ASR) tool to capture every word present in the audio channel. This tool builds a transcription with the precise moment in which every word was pronounced. Then, it is possible to use this transcription to align the subtitles generated by the re-speaker, even those which have no direct correspondence with the audio content (e.g. when the re-speaker summarises a long sentence or several sentences in just a few words). To achieve this, in the core of Deep-Sync, a pre-trained multilingual representation language model based on the deep learning architecture BERT [34] allows to compare at the semantic level the subtitle with the transcription, and based on that, search for the interval of the transcription whose semantic distance to the subtitle is below a certain threshold. When this interval is found, a *transcription-based* alignment is performed, while in those cases where there is no correspondence, a *time-based alignment* is followed.

Additionally, Deep-Sync implements a two-level alignment process. When a transcription-based alignment is performed, the subtitle is synchronised at a sentence level. In order to achieve a perfect positioning of the subtitle, a classic symbol sequence alignment algorithm is then applied at a word level, modifying the presentation time if necessary.

The contributions of this research are summarised below:

- A novel tool for the alignment of the subtitles generated by a re-speaker and the audio-visual content.
- An alignment method between subtitles and transcriptions, made at a semantic level thanks to the use of a deep learning-based language model.
- A second-level alignment allows to perform a fine-grained readjustment at the word level. This allows to make a precise synchronisation of the subtitle.
- An analysis of the alignment results after running Deep-Sync over three sample videos and a comparison against another alignment tool.

The remaining of this article is organised as follows: Sect. 2 summarises the necessary background on TV subtitles services and deep language representation models. Section 3 presents the architecture and operation of Deep-Sync. Section 4 shows the results of running Deep-Sync on three different videos and a comparison against the tool Sub-Sync. Then, Sect. 5 overviews the state of the art on automatic subtitles synchronisation methods and finally, Sect. 6 draws a series of conclusions and a few possible future work lines.

2 Background

2.1 Television subtitle services

The term audio-visual content broadly refers to "any dual media presentation consisting of visual, usually moving, images or pictures, together with auditory, and sometimes printed, language" [35]. Furthermore, subtitling is a special form of audio-visual translation, owing to characteristic features and a number of genre-specific constraints which come into play. These involve spatial, temporal, synchronisation and technical-perceptual issues linked to subtitle distribution and readability [11]. This research is aimed at synchronising the subtitles with the audio-visual contents for live-captioned TV programs by making an automatic adjustment to the time location of both.

Because of its importance in the context of this research, this section details the main characteristics of the different types of TV program along with associated issues regarding the use of accessibility services, such as subtitles.

[1] The source code is available at: https://github.com/alexMyG/deep-sync.

As exposed in [14], television subtitle services can be classified according to the following categories:

- When are they generated (before or during the broadcasting)?
- When are they introduced (in a predefined way, live synchronised, as soon as possible)?
- How are they broadcasted (incrusted, linked, synchronised or unsynchronised)?
- How are they generated (typing, stenography, re-speaking)?

In a pre-recorded program, subtitles can be generated during the editing period or afterwards, regardless of whether insertion is done for all the viewers (open caption) or only visible for those who select the service (closed caption). In those programs, the synchronisation of subtitles and audio-visual content can be done carefully before the broadcasting. In the same way, when the audio-visual production is simultaneous with its broadcast, and the interventions of the speakers are subject to a pre-established script, subtitles' quality is high as long as the speakers comply with said script. In this scenario, the subtitles are broadcasted in a synchronised form, through a procedure that is carried out at the same time of the broadcast that can be automated by means of a wide range of techniques [1, 12, 13, 20].

There is another group of programs where it is not possible to have a rigorous script from which to produce the subtitles, e.g. programs in which guests participate, such as debate or interview programs or those in which the speakers have some freedom to express themselves within the topic to be discussed. Thus, they are improvised broadcasts as they are not bound to a pre-established script, and then, subtitles are generated in real time during the broadcast [12].

One of the most usual techniques for the generation of subtitles in this scenario is the re-speaking. It uses speech recognition technology to obtain a transcription of the locution. It is not feasible under the current state of the art of this technology, to use it directly on the audio of the event [21]. There are several reasons for this. One is that these recognition systems improve their performance significantly when they have been trained for a specific person; this is something that is not controllable in the scenarios to which we refer [10]. However, the most important reason may be that the performance of these systems decreases exponentially according to the level of the background noise (voice, music, other sounds) and is exacerbated when this noise is similar to speech, such as the other speakers' voices on the set [23]. In addition, the speaker needs a minimum amount of time to correct any possible spelling mistakes, to introduce basic punctuation elements and to simplify and summarise the original phrase.

This process introduces a gap or delay between the subtitles and the audio-visual broadcasting. This situation is dysfunctional for multiple reasons. One reason is the confusion created for the hearing-impaired by the lack of correlation between what they are reading and the speaker's lip movements. Another is the confusion produced when subtitles that correspond to an audio of one speaker appear when that speaker is no longer talking and/or the image shows another speaker. Finally, for the hard of hearing, there is the discrepancy between the subtitle and the audio-visual information.

Regarding the presentation of subtitles on the screen, different approaches are depending on the country. In Spain, the subtitles are presented in a block; that is, two complete lines of subtitles appear on the screen at the same time and disappear together. Therefore, in order to show on a screen a subtitle, the time taken by the speaker to complete and correct the sentence and launch it on the screen is added up. The sum of all these times means an average delay of 10 to 20 s from the beginning of the sentence until the subtitle appears on the screen, compromising the synchronisation with the audio and therefore the user's understanding. However, in other countries, it is common to present the subtitle word by word or in a rollup. This approach considerably improves the synchronisation time with the audio by taking less time to present subtitles on the screen.

Finally, in television broadcasting, it is usual that a program is composed by a mix of types (pre-recorded, live-scripted or live-improvised) and there is an additional problem for the subtitle generation and synchrony that arises in the transitions between different types of program. Moreover, in order to recover lost time and gaps, the production control accelerates their release until the natural rhythm of the program is reached. This situation leads to very short subtitle persistence times, which may be too short, even reaching persistence times of zero seconds.

2.2 Deep neural network language models

In the deep learning literature, a language model is a model that has been trained to predict what the next word in a text is, having read the ones before [16]. Since this is a complex task that requires of a semantic understanding of the language, these models are normally trained with massive amounts of data in a self-supervised way, i.e. using labels that are embedded in the data. *BERT* [8], by Google, or *GPT-2*, by OpenAI [31], are examples of popular and successful language models which are extensively used nowadays in many tasks within the field of Natural Language Processing. Normally, trained language models are not used directly. Instead, researchers take an available pre-trained language model as starting point and *fine-tune* it for a task different to what it was originally trained for. This process is known as *transfer learning* [16].

More specifically, in this work we will make use of the pre-trained language model BERT, which stands for “Bidirectional Encoder Representations from Transformers”. BERT makes use of transformer [36], an attention mechanism that learns contextual relations between words (or subwords) in a text. It is designed to pre-train deep bidirectional representations (or embeddings) from unlabelled text by jointly conditioning on both left and right contexts. Once BERT is pre-trained and knows how to represent text, one can use it passing a sequence of words as input. BERT will look left and right several times to that input and produce a vector representation for each word as the output.

Due to the popularity of BERT in the last couple of years, there has been a substantial research effort concerning the creation of different extensions and versions of the model:

- There are two models introduced in the original BERT paper: BERT base, with 110 million parameters, and BERT Large, with 340 million parameters. Thus, training and operating these large models under constrained computational training or inference budgets are challenging. To address these problems, some methods like ALBERT [19] or DistilBERT [34] aim at making the model smaller and faster, while retaining its language understanding capabilities.
- The original BERT model was trained only with English text. However, researchers have replicated its architecture and train it with multiple languages, providing pre-trained models such as M-BERT [30], trained jointly on 104 languages, or BETO [5], trained on a big Spanish corpus.
- While a vanilla BERT can be used for encoding sentences, the embeddings generated with it are not robust. In addition, some sentence-related tasks such as semantic textual similarity require that both sentences are fed into the model, which causes a massive computational overhead and makes BERT unsuitable for this task. Sentence-BERT [32] is a modification of BERT that produces semantically meaningful sentence embeddings that can be then compared using distances like cosine similarity.

From all this versions of BERT, Deep-Sync integrates DistilBERT, which provides a reduced multilingual option for running BERT and has demonstrated excellent results.

3 Deep-Sync

Deep-Sync is a tool for the alignment of subtitles generated during the re-speaking process with the audio-visual content. The core component is a deep learning-based language model, used to encode both the subtitle and the transcription generated in parallel by an automatic speech recognition (ASR) module. Unlike classic alignment methods, the use of a language model allows to take account of the semantic of the words and the sentence as a whole, building an alignment that is resilient to paraphrasing, summarisation, use of synonyms and omission of words. Not only that, the use of these models also offers a wide range of new possibilities. Sentences are represented in a common complex space of language-independent concepts and attributes, enabling to easily extend the model to be used in a multilingual scenario, involving subtitles written in different languages.

The general operation of Deep-Sync is shown in Fig. 1. The content sent from the broadcaster station is received by Deep-Sync, which is integrated in the users’ TV, so the broadcaster does not need to make any changes. The original unadjusted subtitles are separated from the multimedia content, while an ASR captures every word recognised in the audio channel. Meanwhile, the audio-visual content is held 25,000 ms in order to consider the delay introduced by the re-speaker. This allows to ensure that the subtitles associated with the current frames have been received and that the alignment can be performed.

While the subtitles are typically composed by a set of words corresponding to a sentence pronounced by one of the speakers, the ASR operates at a word level, generating a constant flow with every word detected in the audio channel. Deep-Sync calculates the presentation and deletion time of the subtitle with the aid of the transcription, by searching for the sequence of words that better match the last subtitle received (see Fig. 2) and using the timestamps associated with the first and last word in the sequence.

The alignment process starts by receiving two inputs:

- *Subtitles generated during re-speaking* Deep-Sync receives the text of the subtitle together with the presentation and deletion timestamps as defined by the re-speaker. Generally, these timestamps will include a variable delay due to the process of listening and typing the content.
- *Transcription generated by ASR* The ASR generates, in almost real time, a transcription based on the audio content. This operation works at the word level, defining the timestamp associated with each word captured. Given the internal operation of the ASR, words usually show changes as the recording progresses

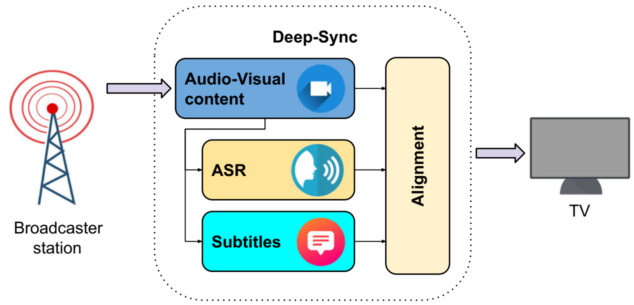

Fig. 1 Diagram showing the general operation of Deep-Sync

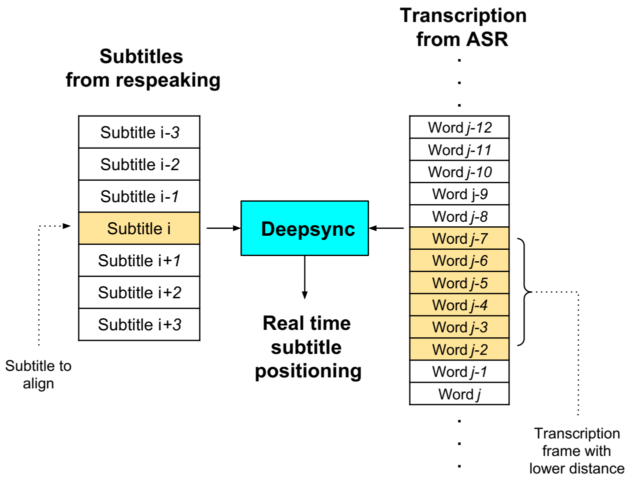

Fig. 2 General diagram showing the alignment procedure of Deep-Sync. On the left, the tool will receive the subtitles as they are created during the re-speaking in real time. At the right, the ASR module will generate a transcription, word by word, of the audio content. Deep-Sync receives both inputs and establishes the most adequate alignment, selecting the frame of words in the transcription which better fits with the subtitle

(e.g. two words which the ASR later considers as one unique word).

3.1 Deep-Sync description

The re-speaking process results often in large differences between subtitles and the transcription. These differences are caused by the use of synonyms, paraphrasing, word and sentence reduction or even complete omission of words by the re-speaker, but also due to errors made by the ASR. To deal with all these facts, Deep-Sync implements the architecture shown in Fig. 3, which includes the following steps:

1. *Subtitle generation* A new subtitle is created by the re-speaker, including the presentation and deletion times. This inevitably introduces a delay which Deep-Sync corrects with the help of the transcription generated with the ASR.
2. *Transcription search interval extraction* A fragment of words within the transcription is extracted, corresponding to the time interval where it is expected to find the best alignment with the subtitle. This interval is, however, still too broad and needs to be curtailed. For this purpose, a search matrix representing all the potential alignments to the subtitle is created.
3. *Similarity checking* The subtitle is iteratively compared against various subsequences of words from the transcription search interval extracted in the previous step. If an alignment is found, which means that there is semantic similarity, Deep-Sync performs a *transcription-based alignment*. If there is no similarity, a *time-based alignment* scheme is used, considering the average of previous delays.
4. *Word-shift adjustment* Due to deliberate omission of words by the re-speaker or to errors during the transcription generation by the ASR, the best alignment found could provoke a word shift between the subtitle and the transcription interval selected. The Needleman–Wunsch algorithm [22] is used to calculate this word shift and to adjust accordingly the presentation and deletion time.
5. *Queued subtitles realignment* Those subtitles already positioned, but not yet sent to the audience due to the fixed broadcasting delay, are readjusted if necessary according to the last subtitles aligned.

The following subsections describe in detail the software architecture, and the most important components of Deep-Sync.

3.2 Transcription-based alignment

In order to align subtitles with the audio-visual content, it is desirable to find a sequence in the transcription generated by the ASR with enough similarity with the subtitle. However, the re-speaking process often introduces significant differences compared to the actual audio content, raising difficulties in finding the subtitle within the transcription. For this reason, the core of Deep-Sync is a semantic comparison module that compares each subtitles against different sequences of words build from the transcription. This procedure provides a useful instrument to obtain the presentation and deletion times of the subtitle even when the texts differ significantly, but they have similar meanings (see Fig. 5).

The transcription generated automatically with the ASR provides a continuous log with every word captured and its timestamp. When a new subtitle s_i is generated and submitted by the re-speaker, it is expected to find a correspondence within a subset of the most recent words of the transcription. In order to identify the specific sequence of

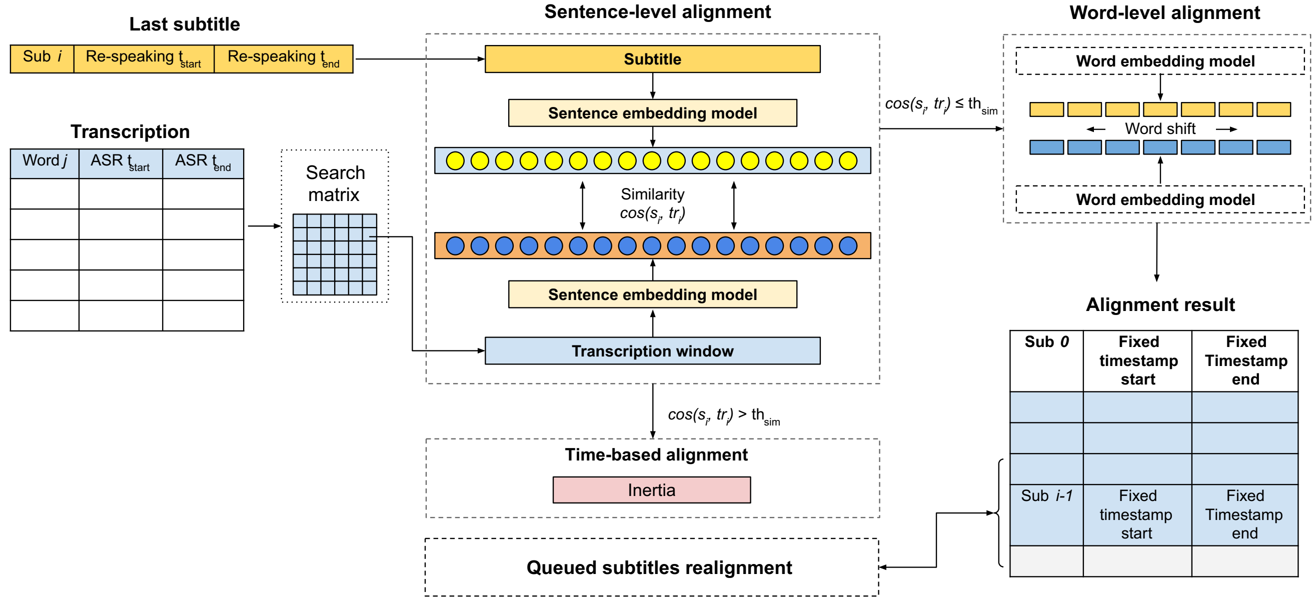

Fig. 3 Diagram showing the architecture of the Deep-Sync tool. The inputs are shown on the left side, including the last subtitle received and the transcription from the ASR. Between the full transcription and the alignment method, a search matrix is used to build potential sentences grouping words from the transcription with different lengths

words with the highest similarity to the subtitle, Deep-Sync first extracts the interval of words between the last word used to align a past subtitle and the last word received from the ASR. Then, by iterating over this search interval, Deep-Sync builds a search matrix that indexes subsequences of words of similar length representing potential alignments. The key idea of this matrix is to create a sliding window with a particular sequence size and to move it throughout the transcription.

3.2.1 Transcription search matrix

Given that the transcription generated with the ASR generates a constant flow of words recognised in real time, an alignment is expected to be found with a subsequence of the last words detected. The transcription search matrix is used to define subsequences of words that are potential alignment candidates. The design of the search matrix is shown in Fig. 4. This matrix defines every subsequence of words which have to be compared against the subtitle. Each cell contains an interval $[w_{\text{start}}, w_{\text{end}}]$ delimiting a specific subsequence of words. Cells in the same row share the index of the first word in the sequence, while cells in the same column share the sequence length. The range of possible values for the sequence length (i.e. the number of columns in the matrix) is calculated according to the subtitle size $\text{size}(s_i)$ (in terms of the number of words), and a threshold θ:

$$l_i = [\text{size}(s_i) - \theta, \text{size}(s_i) + \theta] \tag{1}$$

For each alignment candidate indexed by the search matrix, a semantic comparison against the subtitle allows to measure the level of similarity with it. For this purpose, both the subtitle and the alignment candidate are passed through a deep learning-based language model and transformed to an embedding vector. Deep-Sync then calculates the cosine distance between the two embeddings. The best alignment found among the candidates will be the one with the lowest value of that distance. However, a threshold must be taken into account to consider valid alignments. When the distance between the subtitle and the closer alignment candidate does not fall below the threshold, the transcription does not provide enough certainty and it cannot be used to align the subtitle. In these cases, a time-based alignment is performed as described in Sect. 3.3.

3.2.2 Similarity model

Deep-Sync leverages a sentence embedding library [32] with a pre-trained multilingual model [33] based on a reduced version of BERT [34]. This powerful combination provides a useful representation scheme of sentences at a semantic level. Instead of using a word embedding model, which can only focus on equal words, our proposal takes account of the context and meaning. Besides, this multilingual model has been trained in a plethora of languages: Arabic, Chinese, Dutch, English, French, German, Italian, Korean, Polish, Portuguese, Russian, Spanish and Turkish. Based on this, every sentence is represented in a common space where embeddings with the same semantic will be

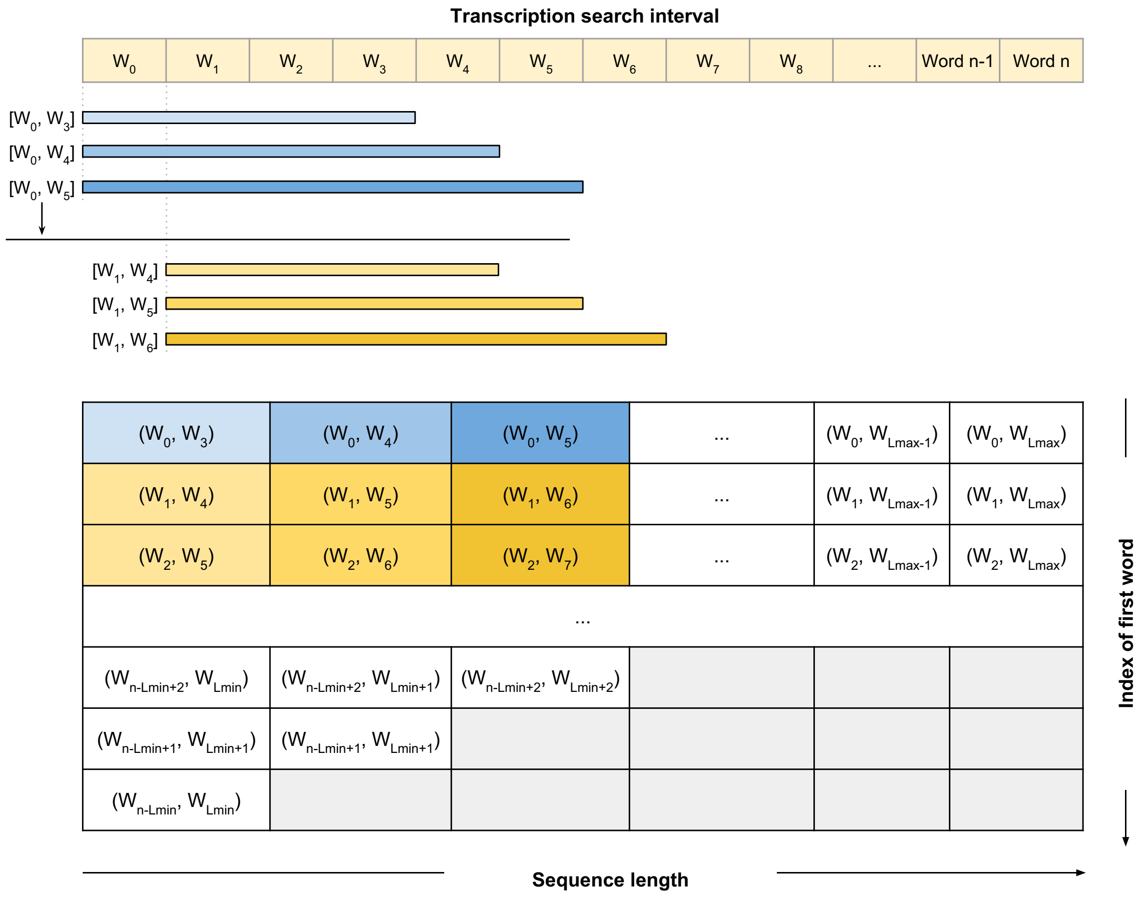
Transcription search interval
W0 W1 W2 W3 W4 W5 W6 W7 W8 ... Word n-1 Word n
[W0, W3]
[W0, W4]
[W0, W5]
[W1, W4]
[W1, W5]
[W1, W6]
(W0, W3) (W0, W4) (W0, W5) ... (W0, WLmax-1) (W0, WLmax)
(W1, W4) (W1, W5) (W1, W6) ... (W1, WLmax-1) (W1, WLmax)
(W2, W5) (W2, W6) (W2, W7) ... (W2, WLmax-1) (W2, WLmax)
...
(Wn-Lmin+2, WLmin) (Wn-Lmin+2, WLmin+1) (Wn-Lmin+2, WLmin+2)
(Wn-Lmin+1, WLmin+1) (Wn-Lmin+1, WLmin+1)
(Wn-Lmin, WLmin)
Index of first word
Sequence length

Fig. 4 Search matrix over the transcription window. Columns refer to the length of the sequence, while rows refer to the index of the first word of the sequence in the transcription

placed close, whatever the language used. In the subtitles domain, this will help to easily align subtitles of different languages.

When comparing a subtitle with a sequence of words from the transcription, their embedding vector is extracted from the language model and the cosine distance (Eq. 2) is used to measure the level of similarity.

$$\cos(\mathbf{a},\mathbf{b}) = \frac{\mathbf{ab}}{\|\mathbf{a}\|\|\mathbf{b}\|} = \frac{\sum_{i=1}^{n} \mathbf{a}_i \mathbf{b}_i}{\sqrt{\sum_{i=1}^{n} (\mathbf{a}_i)^2}\sqrt{\sum_{i=1}^{n} (\mathbf{b}_i)^2}} \tag{2}$$

3.2.3 Word-shift calculation

The search matrix implemented is used to find the best alignment possible between the last subtitle received and the transcription. However, due to words omissions, summarising or small variations, both sequences may entail a word shift, causing the subtitle to be delayed or to be showed ahead of time. For those cases, where this word

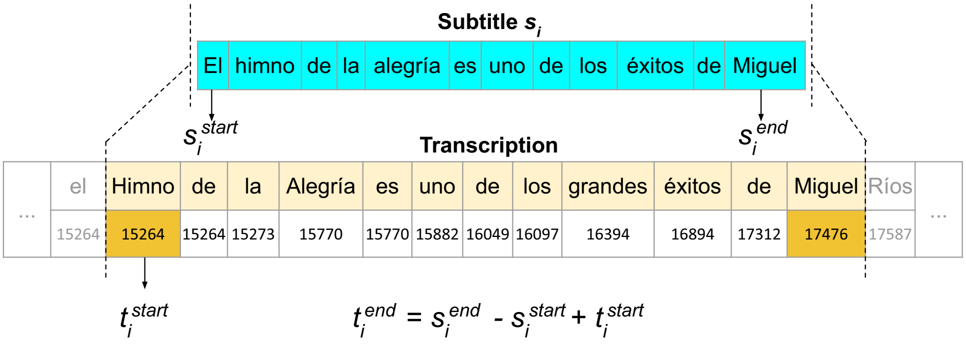
Subtitle s_i
El himno de la alegría es uno de los éxitos de Miguel
s_i^{start}
s_i^{end}
Transcription
... el Himno de la Alegría es uno de los grandes éxitos de Miguel Ríos ...
15264 15264 15264 15273 15770 15770 15882 16049 16097 16394 16894 17312 17476 17587
t_i^{start}
$t_i^{end} = s_i^{end} - s_i^{start} + t_i^{start}$

Fig. 5 Transcription-based alignment

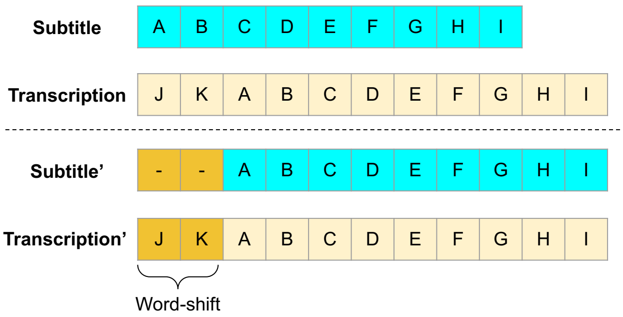

Fig. 6 The Needleman–Wunsch algorithm is used in order to calculate the number of words than the subtitle and the sequence of words extracted from the transcription are shifted

shift involves a delay, the presentation time is adjusted. This is essential in order to avoid delays in subsequent subtitles. This problem can be seen as two desynchronised signals where it is necessary to calculate the time shift between them. For that purpose, first it is necessary to transform each sequence of n words into a vector of n positions. The tokeniser integrated in DistilBERT was used to transform each word into a unique id based on the vocabulary of the model.

Once each sequence is transformed into a chain of tokens, it is possible to apply a symbol sequence alignment algorithm to find the best alignment between both. We applied the Needleman–Wunsch algorithm to build this alignment, which allows to calculate the shift between both chains and to hold back or bring forward the presentation time of the subtitle. Figure 6 shows the result of applying the alignment algorithm over the two chains of symbols.

The preliminary presentation time t_{start} is then adjusted bringing forward the presentation time according to the number of words of displacement and assuming that the average time needed by Spanish speakers to pronounce a word is 375 ms.[2] This value is calculated considering that a Spanish speaker pronounces a maximum of 160 words per minute in a speech [24].

$$t'_{\text{start}} = t_{\text{start}} - \text{word}_{\text{shift}} \cdot 375\,\text{ms} \tag{3}$$

3.3 Time-based alignment

The re-speaking of a broadcasting introduces varied differences between the subtitles and the audio content. Besides, the ASR used to automatically transcribe the audio into words often introduces errors due to several reasons: background noise, multiple individuals speaking simultaneously or difficulties in understanding the speaker. Although the transcription-based alignment (See Sect. 3.2) provides a useful mechanism to locate the subtitle within the transcription, it will not be possible to align those texts when there are drastic differences with between them. In these cases, a time-based alignment is performed, which considers the average delay between the generation time of the subtitle and its associated frame in the transcription. The algorithm employed in this scenario has been replicated from the Sub-Sync tool [14], which the authors called Inertia algorithm. The inertia I_i of a particular subtitle represents the time that it must be brought forward, thus balancing the delay d_i incurred in the re-speaking process.

The inertia algorithm adapted to incorporate the similarity model used by Deep-Sync is defined as follows:

$$cos(s_i, tr_i) < th_{sim} \begin{cases} R_i = s_i - t_i^{start} \\ I_i = 1/2(R_i + I_{i-1}) \end{cases} \tag{4}$$

$$cos(s_i, tr_i) \geq th_{sim} \begin{cases} I_i = I_{i-1} \\ t_i^{start} = s_i - I_i \end{cases} \tag{5}$$

As one can see in the previous equations, the inertia value is calculated when there is alignment by transcription. When there is no alignment, an average of the last delay and previous inertia values is used to position the subtitle.

3.4 Queued subtitles realignment

Although the two previous procedures (transcription and time-based alignments) cover the placement of all subtitles, these methods can incur delays that affect subsequent subtitles. For instance, a bad position calculated by the inertia algorithm could lead to allocate the subtitle with a delay, causing a ripple effect in the next subtitles. For that reason, Deep-Sync includes a queued subtitles realignment process, which only affects to those subtitles that have not been presented yet.

```
$t_{req}$ - Time required to perform realignment
$t_s$ - Start timestamp of a subtitle
$t_e$ - End timestamp of a subtitle
$t_{moved}$ - Last subtitle backward movement
procedure SUBS_REALIGNMENT($align\_list, t_{req}$)
    $subs \leftarrow get\_last\_temp(alignment\_list,\ br\_delay)$
    $t_{moved} \leftarrow 0$
    $t_s, t_e \leftarrow prev\_sub(subs)$
    for sub in subs do
        $t_{moved}, t_s, t_e \leftarrow realign(sub, t_{req}, t_{moved}, t_s, t_e)$
    return $align\_list$
```

More deeply, when the presentation time of a new subtitle is calculated with the transcription-based alignment but it cannot be placed in a given position since it overlaps with a previous subtitle, all previous captions not displayed yet on screen readjusted and moved back in order to build the best global alignment possible. Algorithm 3.4 shows this process. First, the necessary time to position the current subtitle is calculated correctly. Then, each previous

[2] In case Deep-Sync is applied to a different language, this value should be tuned properly.

queued subtitle is moved according to the existing margin between each pair of subtitles and avoiding overlapping and without altering the duration.

3.5 Hardware requirements

When deploying Deep-Sync in a real environment, there is a time interval in which the alignment must be calculated. Thus, it is required to be able to transform each sentence from the transcription and the subtitle to their corresponding embedding vectors. In this case, since we are using a transformer model which entails a considerable amount of parallel computing, a GPU with sufficient memory is needed. In addition, it is also necessary to calculate an important number of distances between each sequence extracted from the transcription and the subtitle. In order to run Deep-Sync properly, we recommend a GPU Titan V 16 GB, similar, or better, at least 16 GB of RAM memory and an Intel Core i5, similar, or better. Since performance will be highly affected by a lack computational, the delay of the broadcasting will have to be adjusted accordingly.

4 Experiments

Deep-Sync has been evaluated using three different sample videos of 30 min each, recorded from different Spanish TV broadcasts. In order to cover most of the possible scenarios thoroughly, the three sample videos contain not only subtitles generated by a re-speaker but also subtitles prepared in advance. With the goal of validating the alignments provided by Deep-Sync, an expert in subtitling visualised every video and repositioned manually the initial and end mark for every subtitle. The difference between these marks and the ones generated by Deep-Sync allows not only to measure the performance of this tool, but also to put it on perspective and make a comparison against Sub-Sync, a subtitle alignment tool based on classic symbol alignment techniques [14]. The three video samples used to run Deep-Sync and Sub-Sync have the following characteristics:

- *Sample video 1* In this sample, almost all subtitles are generated in advance to the broadcast, but are synchronised while the program is on the air.
- *Sample video 2* A sample video with all subtitles generated in real time. The re-speaker generates the subtitles during the live streaming of the broadcast.
- *Sample video 3* A mixed program where half of the subtitle are prepared in advanced and the other half are generated in real time. The type of subtitle switches from one type to the other several times, causing different speeds and high delays changes. This make the alignment process of this video particularly difficult.

The main characteristics' of those videos, including TV channel, program, duration, number of subtitles, etc., are available in the original paper describing Sub-Sync [14]. Every sample video was analysed by the Google Cloud Speech-to-text services (as an ASR),[3] which provides a transcription word by word of the audio. Then, Deep-Sync receives both the subtitles with their original presentation and deletion time and the continuous transcription log generated by the ASR. Different statistics of the results achieved are shown in Table 1. The average start delay in comparison with the reference marks is video samples 2 and 3 below 110ms, a value which can be considered insignificant and will not be noticed by the audience. In sample video 1, this value shows a slight increase, but it is still a low value. As expected, the transcription-based alignment method is the best mechanism to achieve a perfect alignment.

The performance of the alignment process can be appreciated in Figs. 7, 8 and 9. Only the first 200 subtitles of each sample video are displayed to provide a better view. Each bar is related to one subtitle and indicates the difference between the presentation time as generated by Deep-Sync and the presentation time as defined manually by the expert. Most of the subtitles are displayed with a difference lower than 1000ms with the reference timestamps. These deviations are highly imperceptible and involve small differences caused during the generation of the manual references or due to the ASR processing time. It is worth focusing on a specific subset of subtitles in sample video 2, from number 152 to 162. In this time interval, since the re-speaker summarises and reduces several sentences into a lower number, there is no clear position where placing the subtitle. Thus, we can appreciate differences since the expert decided to place the subtitle in a different position in comparison with Deep-Sync.

Figures 10, 11 and 12 show a comparative of the distribution of the delay with respect to the reference timestamps, between the original presentation times and the new timestamps generated by Deep-Sync. In addition, the plots also show the delay depending on the alignment method followed for every subtitle, the transcription-based or the time-based method. Deep-Sync successfully adjusts most of the subtitles in order to make an almost zero delay in most of the subtitles. As it can be seen, the transcription-based alignment method, core of the Deep-Sync tool, is the prime factor behind these results. Nevertheless, those subtitles aligned by the time-based method are also positioned with a huge delay reduction. Table 2 shows a few

[3] https://cloud.google.com/speech-to-text.

Table 1 Summary of the results achieved by Deep-Sync in the three sample videos

	All subtitles				Subtitles aligned with transcription-based method				Subtitles aligned with time-based method			
	Avg. start delay (ms)	Avg. end delay (ms)	% Aligned at start time	% Aligned at end time	Avg. start delay (ms)	Avg. end delay (ms)	% Aligned at start time	% Aligned at end time	Avg. start delay (ms)	Avg. end delay (ms)	% Aligned at start time	% Aligned at end time
Video 1	−538.37	− 37.43	70.06	69.76	−219.02	277.24	94.21	88.95	−1114.22	−770.12	32.67	39.60
Video 2	108.28	784.27	58.08	50.00	− 55.48	654.74	87.34	70.25	432.87	931.62	29.20	27.01
Video 3	−67.28	328.06	83.27	80.00	− 56.52	430.05	94.16	87.80	21.97	45.29	45.56	31.11

The percentage of subtitles correctly aligned is calculated according to a strict threshold of 1000 ms

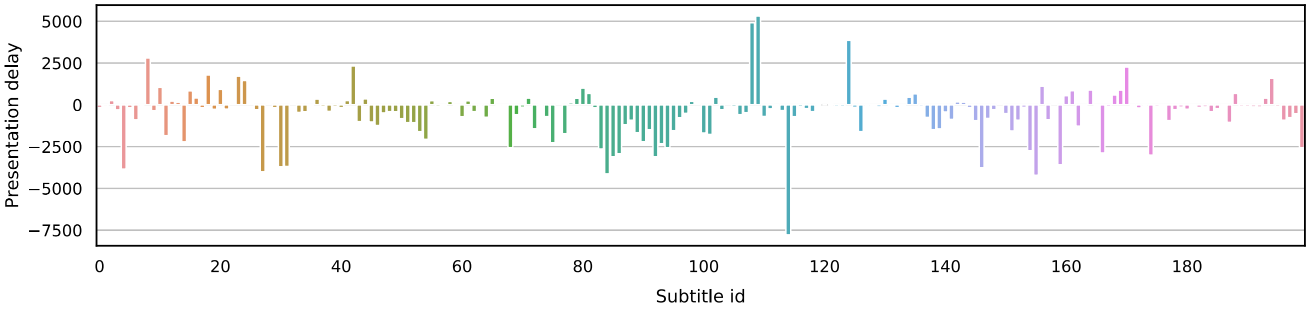

Fig. 7 Difference between presentation time and reference marks for the first 200 subtitle in sample video 1. Values lower 0 mean a delay in the presentation

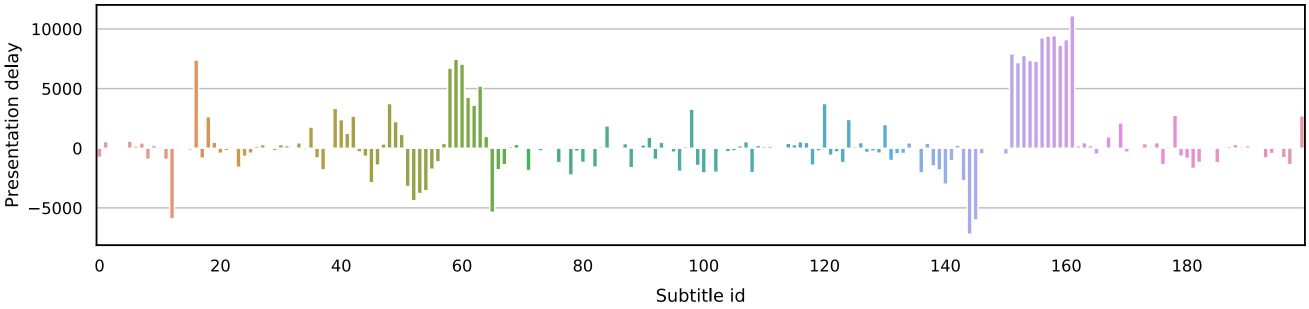

Fig. 8 Difference between presentation time and reference marks for the first 200 subtitle in sample video 2. Values lower 0 mean a delay in the presentation

examples of alignment according to the transcription-based method in sample video 2.

For a better assessment, the delays observed on the alignment generated by Deep-Sync were compared against the tool Sub-Sync. Figures 13, 14 and 15 show improvements in the three sample videos, with higher peaks for zero delay. At the same time, one can appreciate that Sub-Sync tends to cause delays in the presentation time (negative values), a fact which is evident in the plot of the sample video 3 (Fig. 15). In case of sample video 2, an important number of subtitles is presented too early by both Deep-Sync and Sub-Sync. In a few subtitles, this behaviour is due to the repetition of the same sentence, for instance, when the TV presenter introduces a reporter.

A detailed analysis on the number of subtitles better aligned with each method also allows to make a further evaluation. Table 3 shows the number of subtitles for each sample video where Deep-Sync and Sub-Sync achieve a better alignment and the number of subtitles where both methods rise the same delay with respect to the reference timestamps. In the three sample videos, Deep-Sync achieves a better alignment for most of the subtitles in

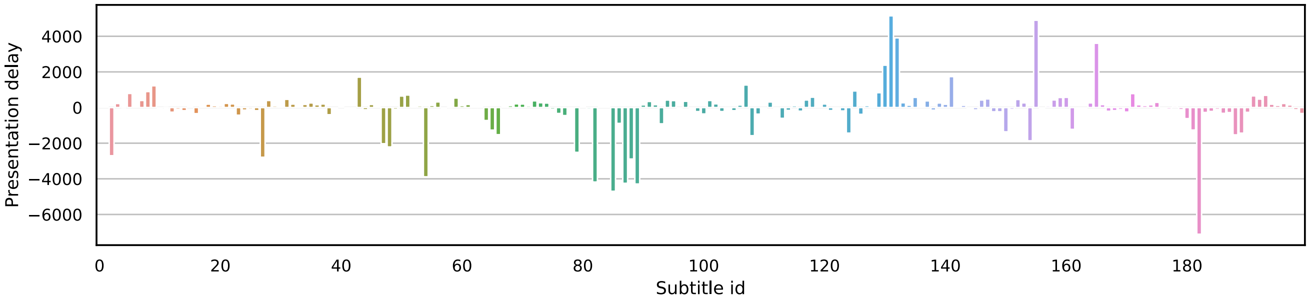

Fig. 9 Difference between presentation time and reference marks for the first 200 subtitle in sample video 3. Values lower 0 mean a delay in the presentation

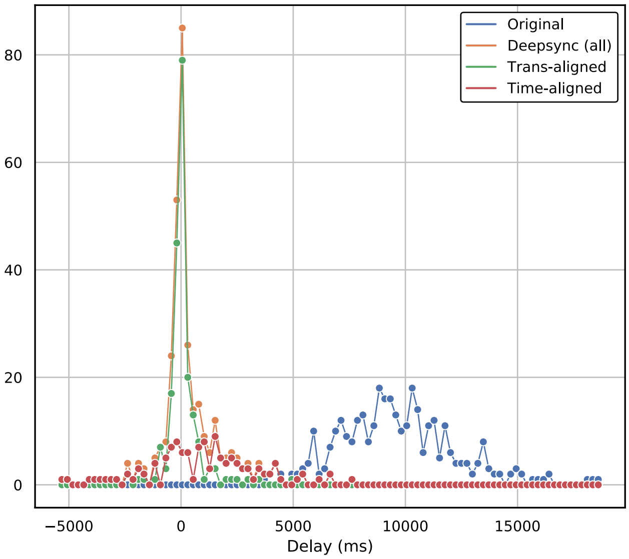

Fig. 10 Delay comparative in sample video 1 between reference timestamps and the original subtitles, the timestamps calculated by running Deep-Sync and the delay generated by each alignment procedure, the transcription-based and the time-based methods

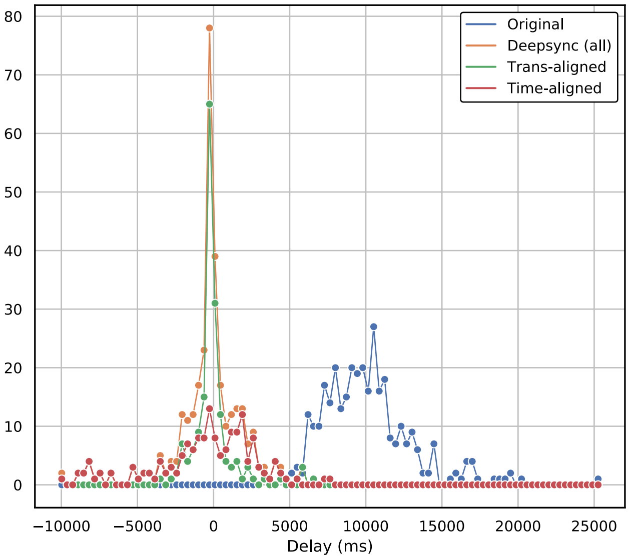

Fig. 11 Delay comparative in sample video 2 between reference timestamps and the original subtitles, the timestamps calculated by running Deep-Sync and the delay generated by each alignment procedure, the transcription-based and the time-based methods

comparison to Sub-Sync. On average, Deep-Sync aligns 9% of the subtitles with lower delay if compared to Sub-Sync.

Finally, we have also measured the time required by Deep-Sync in order to align each subtitle. Since each alignment considers an important number of sequences in the transcription which must be compared against the subtitle, each sequence has to be transformed into its embedding vector. Then, the cosine distance is calculated with the goal of quantifying the distance with the vector representing the subtitle. On average, the embedding of the sequence from the transcription and the calculation of the cosine distance takes around 10ms. Then, for each subtitle it is required more than 80 comparisons to detect the best alignment possible. Thus, it is expected to employ on average one second to perform the alignment, while in some cases, this value can increase significantly when a higher number of comparisons is needed.

5 Related work

Breakthroughs in the automation of live subtitling have involved different researches over the years. This automation is one of the most relevant tasks for improving the process of generating live subtitles. Another parameter that causes significant issues with live subtitles is the synchronisation of them with the audio-visual content. Nowadays, the synchronisation process is currently often done manually and with excessive use of resources. Furthermore, the automatic detection of possible desynchronisation is essential for consumers and broadcasters.

In this context, and over the years, different proposals and algorithms have been compared to find out interesting improvements and insights. Dhumal et al. analysed the capacity of different algorithms to generate subtitles in a bilingual model [9]. Olofsson investigated synchronisation tasks through machine learning approaches [28]. Moreover,

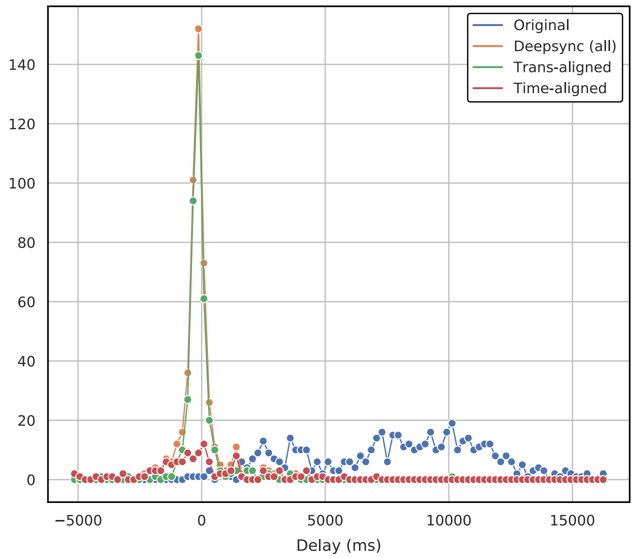

Fig. 12 Delay comparative in sample video 3 between reference timestamps and the original subtitles, the timestamps calculated by running Deep-Sync and the delay generated by each alignment procedure, the transcription-based and the time-based methods

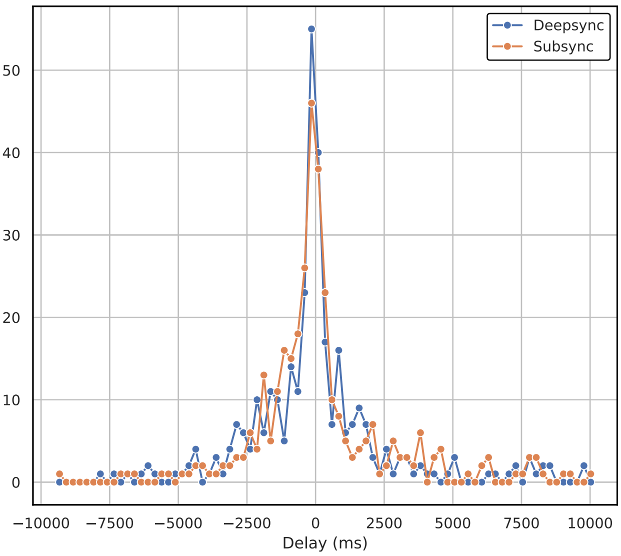

Fig. 14 Delay comparative in sample video 2 between reference timestamps in the alignment generated by Deep-Sync and Sub-Sync

Table 2 Examples of the alignment achieved by Deep-Sync using the transcription-based method in sample video 2

Subtitle	Transcription text
Para ir contextualizando este partido que va a empezar	Contextualizando un poquito lo que es este partido que va a empezar
El Pozo creo que está muy cansado de recoger esa bandeja.	Verdad que muy cansado de recoger la bandeja y le falta ese
No sé si nos da tiempo a escuchar, Roberto, a que escuchemos	Escuchamos no sé si nos da tiempo Roberto a que escuchemos
Ahora no son más de 45. Creo que hemos perdido espectacularidad	45 no más creo que hemos perdido espectacularidad creo que
Es la primera Copa que juego con este nuevo proyecto y equipo.	No primera la primera copa que con este nuevo proyecto y nuevo equipo y

The left column shows the subtitle generated by the re-speaker while the right column shows the range of words selected from the transcription that matches the subtitle. Although the reader could not be familiar with the Spanish language, it can be easily appreciate how words are omitted or how they are located in different places in the subtitle and in the transcription fragment

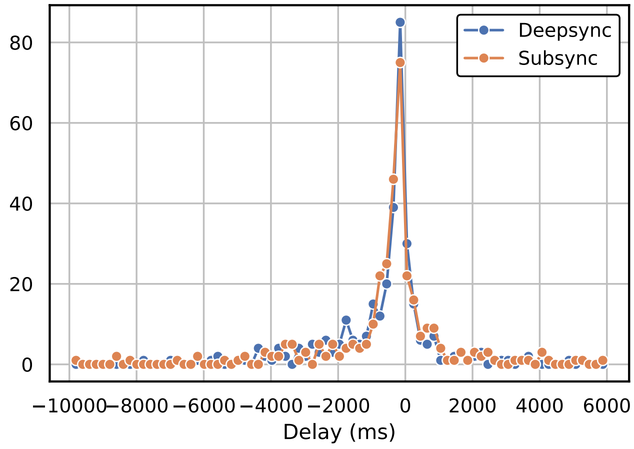

Fig. 13 Delay comparative in sample video 1 between reference timestamps in the alignment generated by Deep-Sync and Sub-Sync

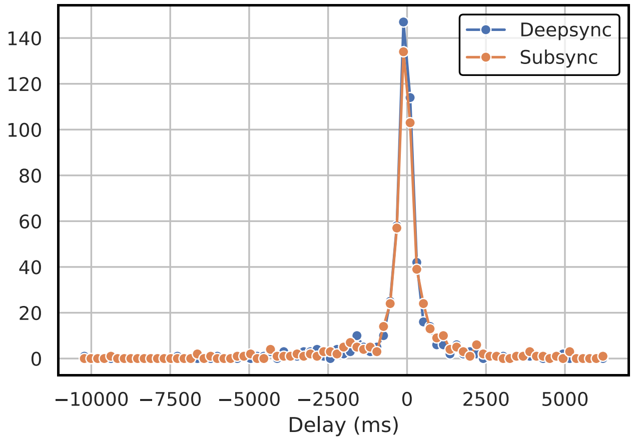

Fig. 15 Delay comparative in sample video 3 between reference timestamps in the alignment generated by Deep-Sync and Sub-Sync

Table 3 Comparison of the number of subtitles better aligned with Deep-Sync when it is compared against Sub-Sync

	Better aligned by Deep-Sync		Better aligned by Sub-Sync		Equal alignment	
	No.	%	No.	%	No.	%
Video 1	129	**38.62**	120	35.93	85	25.45
Video 2	148	**44.33**	122	36.53	64	19.16
Video 3	202	**38.85**	118	22.69	200	38.46

In the three sample videos, Deep-Sync achieves a better alignment for a noticeably higher number of subtitles in comparison with Sub-Sync. In about 25% of the subtitles, the alignment of both methods achieves the same level of accuracy

currently, different researches are focusing on the inclusion of deep learning in ASR components, considering the necessity of a large amount of data for training [29]. This training process has been improved to minimise error rates [3] and to increase the processing systems performance by including more inputs in those components [18].

Synchronisation between audio and text is another point in live captioning that has been studied and improved during past years. For example, in [20], an algorithm to synchronise live speech with its corresponding transcription in real time at the syllabic unit is proposed. The goal of this research is to apply the algorithm for generating audio books in unstructured language like Thai from live speech. Moreover, the desynchronisation, also called latency, is much higher in those ASR of small queries, such as Google Assistant or Apple Siri, and it is not usable in longer speeches [26].

While synchronisation is a complicated goal to achieve due to the state of technology, other improvement proposals tend to dynamic adjustments in the display of subtitles, looking for a re-programming where these subtitles are already synchronised [2]. When re-programming is the starting point, the user experience is improved from the point of view of personalisation and synchronisation, which is usually done dynamically [25], although this synchronisation is not strict, as indicated by the quality parameters [15] [4].

In a real environment with live subtitling, such as television, the changes in consumption and technology have improved the scenario in a short time. The arrival of broadband hybrid television (HbbTV) generated an expectation of improvement in this synchronisation and reprogramming of subtitle flows. In [17], a three-module algorithm is proposed in HbbTV context for automatic synchronisation of subtitle and video content. Moreover, media synchronisation (e.g. broadcast video with subtitles or alternative audio received via the Internet) is receiving renewed attention with ecosystems of connected devices enabling novel media consumption paradigms [7].

The possibility of using Natural Language Processing (NLP) for parameter customisation has been also analysed. In [6], an interactive system aimed at automatically generating video summaries and performing subtitles synchronisation for persons with hearing loss is presented. The module that generates the video summaries uses techniques such as latent semantic analysis (LSA) and latent Dirichlet allocation (LDA), whereas the synchronisation module is based on forced alignment between audio streams and text.

Finally, it is worth to mention that deep learning-based language models such as BERT, which are the base in which Deep-Sync is built on, have been recently applied in a wide variety of NLP tasks, such as sentiment analysis and question answering, surpassing the performance of previous approaches in the field [8]. In addition, it is remarkable that, starting from October 2019,[4] Google Search started to roll out BERT to better understand the searches of users, and nowadays, BERT is used on almost every English query made on this popular service.

6 Conclusion

Deep-Sync has been designed aiming to provide a practical instrument to reach a better experience for those who make use of subtitles while watching TV. In contrast to other approaches in the state of the art, our proposal is built on an deep learning-based language representation model. This allows to consider the semantic of the subtitle when looking for a match in the transcription generated by the ASR, aligning correctly most of the captions. Besides, Deep-Sync integrates a word-level readjustment method, based on the Needleman–Wunsch algorithm, which calculates the word shift between the subtitle and the alignment candidate found in the transcription. The results achieved after running Deep-Sync with three different sample videos demonstrate that this tool can be successfully deployed in mixed (usually complicated) scenarios, to align subtitles generated in real time by a re-speaker but

[4] https://blog.google/products/search/search-language-understanding-bert/.

also when they are generated in advance. Future work lines involve improving the alignment of the subtitles for which there is no correspondence in the transcription and testing other automatic speech recognition tools.

Acknowledgements This work has been supported by the Spanish Ministry of Science and Education under TIN2017-85727-C4-3-P grant (DeepBio) and Comunidad Autónoma de Madrid under S2018/TCS-4566 grant (CYNAMON). We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan V GPU used for this research.

Compliance with ethical standards

Conflict of interest The authors declare that there is no conflict of interests regarding the publication of this paper.

Code availability The code of Deep-Sync and the instructions to be executed are available at: https://github.com/alexMyG/deep-sync.

References

1. Ando A, Imai T, Kobayashi A, Homma S, Goto J, Seiyama N, Mishima T, Kobayakawa T, Sato S, Onoe K et al (2003) Simultaneous subtitling system for broadcast news programs with a speech recognizer. IEICE Trans Inf Syst 86(1):15–25
2. Avegliano PB, Real LCV, Guimaraes RL, Gallo DS (2017) Automatic synchronization of subtitles based on audio fingerprinting. US Patent 9,609,397
3. Baskar MK, Burget L, Watanabe S, Karafiát M, Hori T, Černockỳ JH (2019) Promising accurate prefix boosting for sequence-to-sequence asr. ICASSP 2019–2019 IEEE international conference on acoustics. Speech and signal processing (ICASSP), IEEE, pp 5646–5650
4. Brito JO, Santos CA, Guimarães RL, Borges TFC (2019) Toward understanding the quality of subtitle synchronization to improve the viewer experience. In: Proceedings of the 25th Brazillian symposium on multimedia and the web, pp 209–216
5. Cañete J, Chaperon G, Fuentes R, Pérez J (2020) Spanish pre-trained bert model and evaluation data. In: to appear in PML4DC at ICLR 2020
6. Cuzco-Calle I, Ingavélez-Guerra P, Robles-Bykbaev V, Calle-López D (2018) An interactive system to automatically generate video summaries and perform subtitles synchronization for persons with hearing loss. 2018 IEEE XXV international conference on electronics. Electrical engineering and computing (INTERCON), IEEE, pp 1–4
7. van Deventer MO, Stokking H, Hammond M, Le Feuvre J, Cesar P (2016) Standards for multi-stream and multi-device media synchronization. IEEE Commun Mag 54(3):16–21
8. Devlin J, Chang MW, Lee K, Toutanova K (2018) Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv:181004805
9. Dhumal M, Kushwaha HK, Gupta V, Pawara SR (2019) Instant bi-lingual captions. In: 2019 IEEE 5th international conference for convergence in technology (I2CT), IEEE, pp 1–6
10. Gales MJ (2001) Adaptive training for robust ASR. In: 2001 IEEE workshop on automatic speech recognition and understanding, ASRU 2001 - conference proceedings, IEEE, pp 15–20, https://doi.org/10.1109/ASRU.2001.1034578
11. Gambier Y (2003) Introduction: Screen transadaptation: Perception and reception. The Translator 9(2):171–189. https://doi.org/10.1080/13556509.2003.10799152
12. Gao J, Zhao Q, Li T, Yan Y (2009) In: International symposium on neural networks. Simultaneous synchronization of text and speech for broadcast news subtitling. Springer, pp 576–585
13. Garcia JE, Ortega A, Lleida E, Lozano T, Bernues E, Sanchez D (2009) Audio and text synchronization for tv news subtitling based on automatic speech recognition. In: 2009 IEEE international symposium on broadband multimedia systems and broadcasting, IEEE, pp 1–6
14. González-Carrasco I, Puente L, Ruiz-Mezcua B, López-Cuadrado J (2019) Sub-sync: Automatic synchronization of subtitles in the broadcasting of true live programs in spanish. IEEE Access 7:60968–60983
15. Guimarães RL, Brito JO, Santos CA (2018) Investigating the influence of subtitles synchronization in the viewer's quality of experience. In: Proceedings of the 17th Brazilian symposium on human factors in computing systems, pp 1–10
16. Howard J, Gugger S (2020) Fastai: A layered api for deep learning. Information 11(2):108
17. Kedačić D, Herceg M, Peković V, Mihić V (2018) Application for testing of video and subtitle synchronization. In: 2018 International conference on smart systems and technologies (SST), IEEE, pp 23–27
18. Krishnamoorthy M, Paulik M (2019) Automatic speech recognition based on user feedback. US Patent 10,446,141
19. Lan Z, Chen M, Goodman S, Gimpel K, Sharma P, Soricut R (2019) Albert: A lite bert for self-supervised learning of language representations. arXiv:190911942
20. Lertwongkhanakool N, Punyabukkana P, Suchato A, (2013) Real-time synchronization of live speech with its transcription. In: 10th international conference on electrical engineering/electronics, computer, telecommunications and information technology, IEEE, pp 1–5
21. Li J, Deng L, Haeb-Umbach R, Gong Y (2016) Fundamentals of speech recognition. Robust Automatic Speech Recognition pp 9–40, https://doi.org/10.1016/b978-0-12-802398-3.00002-7, 1001.2267
22. Likic V (2008) The needleman-wunsch algorithm for sequence alignment. In: Lecture given at the 7th Melbourne Bioinformatics Course, Bi021 Molecular Science and Biotechnology Institute, University of Melbourne pp 1–46
23. Maas AL, Le QV, O'Neil TM, Vinyals O, Nguyen P, Ng AY (2012) Recurrent neural networks for noise reduction in robust ASR. In: 13th Annual conference of the international speech communication association 2012, INTERSPEECH 2012, vol 1, pp 22–25
24. Manuel Jerez JAd (2005) La incorporación de la realidad profesional a la formación de intérpretes de conferencias mediante las nuevas tecnologías y la investigación-acción. http://hdl.handle.net/10481/871
25. Montagud M, Boronat F, González J, Pastor J (2017) Web-based platform for subtitles customization and synchronization in multi-screen scenarios. In: Adjunct publication of the 2017 ACM international conference on interactive experiences for TV and online video, pp 81–82
26. Nguyen TS, Niehues J, Cho E, Ha TL, Kilgour K, Muller M, Sperber M, Stueker S, Waibel A (2020) Low latency asr for simultaneous speech translation. arXiv:200309891
27. Ofcom (2005) Subtitling–an issue of speed?
28. Olofsson O (2019) Detecting unsynchronized audio and subtitles using machine learning
29. Park DS, Chan W, Zhang Y, Chiu CC, Zoph B, Cubuk ED, Le QV (2019) Specaugment: A simple data augmentation method for automatic speech recognition. arXiv:190408779

30. Pires T, Schlinger E, Garrette D (2019) How multilingual is multilingual bert? arXiv preprint arXiv:190601502
31. Radford A, Wu J, Child R, Luan D, Amodei D, Sutskever I (2019) Language models are unsupervised multitask learners. OpenAI Blog 1(8):9
32. Reimers N, Gurevych I (2019) Sentence-bert: Sentence embeddings using siamese bert-networks. In: Proceedings of the 2019 conference on empirical methods in natural language processing, association for computational linguistics, http://arxiv.org/abs/1908.10084
33. Reimers N, Gurevych I (2020) Making monolingual sentence embeddings multilingual using knowledge distillation. arXiv:200409813 http://arxiv.org/abs/2004.09813
34. Sanh V, Debut L, Chaumond J, Wolf T (2019) Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. 1910.01108
35. Souto-Rico M, González-Carrasco I, López-Cuadrado JL, Ruíz-Mezcua B (2020) A new system for automatic analysis and quality adjustment in audiovisual subtitled-based contents by means of genetic algorithms. Expert Syst. https://doi.org/10.1111/exsy.12512
36. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I (2017) Attention is all you need. In: Advances in neural information processing systems, pp 5998–6008