Behavior Research Methods (2018) 50:466–489
https://doi.org/10.3758/s13428-017-1002-7

# AlignTool: The automatic temporal alignment of spoken utterances in German, Dutch, and British English for psycholinguistic purposes

Lars Schillingmann[1] · Jessica Ernst[2] · Verena Keite[2] · Britta Wrede[1] · Antje S. Meyer[3,4] · Eva Belke[2]

Published online: 29 January 2018

## Abstract

In language production research, the latency with which speakers produce a spoken response to a stimulus and the onset and offset times of words in longer utterances are key dependent variables. Measuring these variables automatically often yields partially incorrect results. However, exact measurements through the visual inspection of the recordings are extremely time-consuming. We present AlignTool, an open-source alignment tool that establishes preliminarily the onset and offset times of words and phonemes in spoken utterances using Praat, and subsequently performs a forced alignment of the spoken utterances and their orthographic transcriptions in the automatic speech recognition system MAUS. AlignTool creates a Praat TextGrid file for inspection and manual correction by the user, if necessary. We evaluated AlignTool's performance with recordings of single-word and four-word utterances as well as semi-spontaneous speech. AlignTool performs well with audio signals with an excellent signal-to-noise ratio, requiring virtually no corrections. For audio signals of lesser quality, AlignTool still is highly functional but its results may require more frequent manual corrections. We also found that audio recordings including long silent intervals tended to pose greater difficulties for AlignTool than recordings filled with speech, which AlignTool analyzed well overall. We expect that by semi-automatizing the temporal analysis of complex utterances, AlignTool will open new avenues in language production research.

**Keywords** Language production · Time course · Voice-key · Automatic alignment

## Introduction

In light of the fluency and swiftness with which speakers produce spoken language (see Levelt, 1989), many studies of language production are concerned with the temporal coordination of utterance encoding processes. The dependent variables are the latency with which speakers produce a spoken response to a stimulus or the temporal structure of spoken utterances (specifically onset- and offset-times of words). For instance, in order to study the retrieval of words from the mental lexicon, researchers often use picture naming tasks requiring speakers to retrieve the names of objects shown on the screen. The temporal onsets of the spoken words are taken as the dependent variable, as they reflect the time taken for retrieving and preparing the response (Bock, 1996; Griffin & Ferreira, 2006; Levelt, Roelofs, & Meyer, 1999). When seeking to study the formulation of more complex utterances, researchers often use eye-tracking technology to study where speakers look while describing a scene or reading a text (for a review, see Hüttig, Rommers, & Meyer, 2011, and Rayner, 1998, respectively). A dependent variable of particular interest is the eye-voice-span, that is, the time elapsing between the onset of a gaze on an object or a word on the screen and the onset of the participant's naming response (e.g., Griffin & Bock, 2000; Laubrock & Kliegl, 2015).

For measuring onset times of single words in response to a stimulus, many researchers use custom-made hardware or software tools (often called "voice-keys") that establish the onset times of the first acoustic input after stimulus presentation (Rastle & Davis, 2002). However, as we will review shortly, voice-keys do not provide consistently accurate measurements and can be triggered by non-speech signals preceding the response. In addition, they measure the onsets of

✉ Eva Belke
belke@linguistics.rub.de

[1] Technische Fakultät, Universität Bielefeld, Bielefeld, Germany

[2] Sprachwissenschaftliches Institut, Ruhr-Universität Bochum, Bochum, Germany

[3] Max-Planck-Institut für Psycholinguistik, Nijmegen, The Netherlands

[4] Radboud University, Nijmegen, The Netherlands

utterances only. When having to analyze the full temporal structure of multiple-word utterances, many psycholinguists resort to establishing the onset and offset times of individual words by visually inspecting the waveforms of the utterances in standard audio editors, such as Praat (Boersma & Weenink, 2016) or Audacity (Audacity team, 2016). This is an extremely time-consuming procedure. In addition, the reliability of the hand-measurements is often unknown.

We present AlignTool, a tool for semi-automatically analyzing the full temporal structure of complex utterances, including utterance onsets and offsets as well as the onsets and offsets of individual words and phonemes in utterances. We demonstrate that for high quality recordings with an excellent signal-to-noise ratio, AlignTool can work in a largely automatic mode requiring virtually no corrections by the user. With recordings of lesser quality, the automatic measurements AlignTool establishes are less exact and require manual corrections.[1] However, AlignTool is still functional when working with such recordings, and it provides researchers with useful hypotheses of where to set relevant time stamps and hence helps to reduce the manual annotation time. By generating TextGrid files, AlignTool allows users to easily adjust individual time stamps using Praat. Note, however, that AlignTool does not work for audio files with substantial background noise, such as speech recordings made in an MRI scanner.

In being functional even for recordings of lesser quality, AlignTool presents a promising bridge between the two major research traditions in language production research (Clark, 1996). The *language-as-product* (or *chronometric*; Levelt, 1999) tradition is the research tradition we have outlined above. It focuses on the cognitive representations and processes underlying utterance generation. Typically, research in this tradition involves testing individual speakers in a laboratory situation. In the *language-as-action* tradition, by contrast, linguistic behavior is studied in the context of its natural dialogue context. For instance, researchers may be interested in the length of pauses in speaking (e.g., Clark, 1996; Marklund, Marklund, Lacerda, & Schwarz, 2015) or in the duration of individual words (Fox Tree & Clark, 1997; Mousikou & Rastle, 2015). This requires that the temporal onsets and offsets of words be established. Note, however, that establishing as natural dialogue settings as possible (including persons moving freely in a room or even on a university campus; Brennan, Schuhmann & Batres, 2013), often comes at the expense of losing the technical precision required for high-quality audio recordings with a near-perfect signal-to-noise ratio.

Eye-tracking technology is already starting to build a bridge between the two research traditions (for early examples, see Hanna & Brennan, 2007; Metzing & Brennan, 2003), as it allows researchers to assess with high temporal resolution which information in the environment speakers and listeners are attending to while listening to or generating utterances (e.g., Coco & Keller, 2015; Coco, Malcolm & Keller, 2014). In order to link this information to the time course of linguistic processes in production and perception, the temporal structure of the spoken utterances must be analyzed at a similar level of granularity. To date, this is extremely time-consuming as the spoken utterances are mostly labelled manually. AlignTool was designed to automatize large portions of these analyses, reducing effective annotation time by up to two thirds. This estimate is based on the time it took our trained annotators to analyze an audio file of good recording quality manually; with audio data of lesser quality, the reduction in annotation time might be less pronounced. We expect that by rendering the temporal analyses of more complex utterances feasible, AlignTool will aid in bridging the gap between the language-as-product and the language-as-action traditions.

## Establishing the temporal structure of speech for psycholinguistic purposes

### Establishing utterance onsets

To our knowledge, the only domain that has been automatized to some extent and with mixed success is the measurement of utterance onsets. Most researchers use a custom-made hardware or software voice-key, which can be programmed to measure the time that elapses between the presentation of a stimulus and the first audio input to the participant's microphone, registered physically as sound pressure. Hardware-based voice-keys typically convert the acoustic energy recorded at the participant's microphone into electric energy and are triggered as soon as the energy (i.e., the sound pressure) exceeds a pre-specified threshold. Alternatively, software-based voice-keys have been developed that measure response times based on an algorithmic analysis of audio files (e.g., Jansen & Watter, 2008; Protopapas, 2007).

The language production research community has long been aware that voice-keys of this type are potentially inaccurate. For instance, voiced word onsets (as in *man*) are likely to be detected better than unvoiced onsets (as in *fall*); voiceless fricative onsets (as in *fall*) are likely to yield more variable measures than voiceless plosives (as in *tall*; Duyck, Anseel, Szmalec, Mestdagh, Tavernier, & Hartsuiker, 2008; Kessler, Treiman, & Mullenix, 2002; Pechmann, Reetz, & Zerbst, 1989; Rastle & Davis, 2002). Kessler et al. (2002; see also Pechmann et al., 1989) had participants read out 2,982 words as quickly and as accurately as possible in a speeded naming task, establishing response times by means

[1] In Belke et al. (2017), we give some practical advice on how to obtain good audio recordings (see Section 5, Step 0).

of a voice-key. When all item-related variables (familiarity, frequency, length) except for the initial phoneme were entered into a multiple regression analysis of the response times, they accounted for 35 % of the variance. Further analyses showed that the residual reaction times were significantly affected by the identity of the initial phoneme of the word, suggesting that there is a substantial phonetic bias in voice-key measurements. Findings like these have led language production researchers to adapt their experimental designs in order to minimize potential voice-key-related confounds, for instance by matching the stimuli used in different experimental conditions with regard to their word onsets.

However, this practice may not suffice because not only initial phonemes but second phonemes, too, appear to impact on the accuracy of the voice-key measurements (see Kessler et al., 2002). A particularly dramatic demonstration of this problem was reported by Rastle and Davis (2002), who had 24 participants name 40 monosyllabic words beginning with the phoneme /s/. Half of these words had a CVC structure with simple onsets (e.g., *sat*), the other half had a CCVC structure with complex onsets (e.g., *spat*). In addition, participants read aloud filler words, none of which started with /s/, which were not considered in the analyses. Naming latency measurements were obtained from two voice-keys and from a human rater who visually inspected the waveforms of all responses. The two voice-keys differed marginally in that the first one, which was the built-in voice-key of a frequently used experimental control software (DMDX; see L.I. Forster & J.C. Forster, 2003), read out the amplitude of the input registered at the sound card and triggered as soon as the amplitude moved above (below) a pre-defined upper (lower) threshold. The second voice-key was a custom-made voice-key constructed at the University of Cambridge (see also M. D. Tyler, L. Tyler & Burnham, 2005). It was not only sensitive to the amplitude of the signals but also to their duration. This way, the voice-key was not only triggered by high-intensity signals but potentially also by low-intensity signals that are sufficiently long-lasting (e.g., /s/ in the example). The results showed that the voice-keys established the speech onset approximately 100 ms later than the human rater; more disturbingly, however, the reaction time difference between the words with simple and complex onsets differed substantially between the measurement techniques: Hand-marked measurements yielded longer latencies for simple than for complex onsets, while the measurement of the DMDX-voice-key yielded longer latencies for complex than for simple onsets. The measurements obtained from the custom-built voice-key yielded no difference between simple and complex onsets.

The results reported by Rastle and Davis (2002) call into question the validity of voice-key-based measurements. Partly in response to this, several research groups have attempted to develop highly accurate digital voice-keys for language production research in the past 10 years:

- Protopapas (2007) developed CheckVocal, a program specifically designed to re-edit measurements and recordings made using the experimental control software DMDX (L.I. Forster & J.C. Forster, 2003). CheckVocal allows users to manually adjust measurements performed by the built-in voice-key of DMDX.
- Jansen and Watter (2008) presented SayWhen, a software-based voice-key that operates on recordings of full experimental sessions, independently of the experimental control software that they were originally recorded with. It requires that experimenters record an audio-signal at the moment of stimulus onset in order for the software to compute response latencies with reference to the stimulus onset. Hence, like CheckVocal, SayWhen operates on the recorded experimental session off-line, which provides the opportunity for the experimenter to revisit those trials that may be hard to measure for the software. SayWhen includes a problem-tagging component that tells the user which trials were hard to analyze and may need to be inspected visually for accurate measurements of response latencies. With both CheckVocal and SayWhen, the combination of an automatic analysis of unambiguous trials along with the possibility for manual corrections of ambiguous trials ensures that the proportion of trials that have to be discarded for measurement problems is minimal.
- Abrams and Jennings (2004) developed a software, VoiceRelay, that uses a purely amplitude-based threshold criterion to establish voice onsets in experiments with reference to the presentation of a stimulus. In light of the disadvantages of purely amplitude-based voice-keys that we have discussed above (see Rastle & Davis, 2002), VoiceRelay appears to be less promising than the approach presented by Jansen and Watter (2008).
- Roux et al. (2016) presented Chronset, a fully automated alignment tool that makes use of multiple acoustic features of the signal (such as spectral change, amplitude modulation, and frequency modulation). For an onset to be detected, four of six acoustic features must exceed a predefined threshold for at least 35 ms. The thresholds were established based on manually annotated datasets in Spanish and English (taken from Sadat, Martin, Alario, & Costa, 2012, and Jansen & Watter, 2008, respectively). In order to render the thresholds valid and applicable to other languages than Spanish and English, the authors carried out a regression analysis, minimizing the standard deviation of the residuals of the six acoustic features. The resulting thresholds proved reliable when used with data from a different language and a different recording environment, promising broad generalizability of Chronset across languages. However, Chronset is unlikely to achieve perfect accuracy. Roux et al. (2016) used Monte Carlo simulations to demonstrate that for sample sizes of 23 or more, simulated measurement errors with

> a standard deviation of 87 ms led to a reduction in statistical power of less than 10 %, decreasing further with larger sample sizes (p. 15). However, for some users, this level of accuracy may be unacceptable, especially in light of the fact that Chronset does not allow for a post-hoc manual correction of the measurements.

While Chronset's accuracy is promising, it is clear from this review that, to date, no fully automatic and fully reliable voice-key has been engineered. Hence, it would seem that the most promising way forward is to design a tool that provides a good preliminary temporal analysis of the audio data and that allows the user to easily correct the measurements where necessary. AlignTool was designed to achieve this goal, reducing dramatically the work load associated with manual measurements of the onset and offset times of words and phonemes in utterances.

### Beyond utterance onsets

As discussed previously, many eye-tracking studies make use of scene description or reading tasks that require the participants' utterances to be annotated temporally beyond the utterance onset so as to be able to link the temporal structure of their eye movements to that of their speech. Similarly, for researchers interested in the comprehension of spoken language, it may be of interest to link the temporal structure of a spoken utterance processed by participants to their eye movements. Finally, in dialogue these two perspectives are linked. Continuous speech can be aligned temporally with a literal transcription using automatic speech recognition in a procedure called forced alignment. Users feed transcripts of the spoken utterances to the speech recognition system and make it align the speech into the audio signal. MAUS (Schiel, 1999, 2015), an acronym for Munich Automatic Segmentation System, is a tool that allows users to feed the recordings of spoken utterances and their literal transcriptions into the automatic speech recognition system HTK, which then maps what has been said onto the speech signal (forced alignment; see Appendix 1) and generates a TextGrid file specifying the onset and offset times of words and phonemes. WebMAUS (Kisler, Reichel, Schiel, Draxler, Jackl, & Pörner, 2016) presents a web-based interface for using MAUS (see Rosenfelder, Fruehwald, Evanini, Keelan, & Jiahong, 2011, for a similar application for the English language only).

## AlignTool

AlignTool was designed to cover the full range of functionalities reviewed so far. It was developed in Python under Linux and runs under Windows within a virtual Linux environment. After a pre-segmentation of the audio signal with respect to speech onset and offset times using Praat (Boersma and Weenink, 2016), AlignTool uses the automatic speech recognition system MAUS (Schiel, 1999, 2015) via WebMAUS (Kisler et al., 2016) to force-align transcripts of the utterances by the speakers with the speech signal (Strunk, Schiel & Seifart, 2014). To this end, an orthographic transcription provided by the user is transformed into a phonotypical transcription by means of a phonetic lexicon. Based on this phonotypical transcription, a sequence of acoustic phone models is derived, which is force-aligned with the speech signal, providing the most likely temporal alignment of speech and transcription (see Appendix 1). By creating TextGrid files as an output, AlignTool allows its users to manually correct the results of its analyses in the TextGrid file, if necessary. This may be particularly helpful for analyses of audio data of poor quality. All manual changes to the TextGrid file can be saved and the corrected data exported to an Excel file (or equivalent) for further data processing.

AlignTool can analyze recordings of single trials but also recordings of whole experiment blocks, i.e., series of trials, provided that the recording contains segmentation signals indicating trial onset times. AlignTool handles single word utterances but also multiple word utterances and semi-spontaneous speech in dialogue-like experimental settings, such as the Map Task (Anderson et al., 1991). Note that for analyzing single- or multiple-word utterances in experimental settings, it is indispensable to first apply AlignTool's pre-segmentation routine. It establishes the location of the speech interval in the recording of a given trial, filtering out the silent intervals at the beginning and the end of the recording. In earlier work with MAUS (Schiel, 1999) and ESMERALDA (Fink, 1999; Katzberg, Belke, Wrede, Ernst, Berwe, & Meyer, 2014), we have found that long silent intervals are difficult to accommodate for automatic speech recognition systems, most likely due to the lack of (phonetic) structure in these intervals. Therefore, AlignTool makes use of Praat to detect the beginning and end of the spoken utterance in a trial and subsequently force-aligns the pre-segmented portions of the audio signal using MAUS.

To date, AlignTool is able to handle German, Dutch, and British English speech input. An extension to other languages is possible, as the only language-sensitive processing step of the tool relies on WebMAUS, which supports other languages, such as Italian, Spanish, Russian, as well as other variants of English and German (Australian and American English, Swiss German etc.).

Our aim was to design AlignTool for average Windows users with little programming experience. To this end, we embedded the Linux environment required for working with MAUS into a virtual environment that can be run under Windows. Users interact with AlignTool via an Excel file[2] and hence are not required to use the Linux command line.

[2] Alternatively, users can use OpenOffice/LibreOffice. For the sake of consistency with the User Manual (Belke, Keite, & Schillingmann, 2017) we refer to Excel in the following.

At the same time, we give users with more programming expertise the opportunity to improve and expand AlignTool and therefore make it available open-source (see Author note).

## System requirements and terms of usage

AlignTool runs under Linux Ubuntu natively. In order to use it with a Microsoft Windows operating system, it is necessary to simulate a virtual Linux environment. For this purpose, VMWare Workstation Player is required. AlignTool runs as a Virtual Machine in the Workstation Player. Note that VMware Workstation Player requires a 64-bit Windows processing system and, in some cases, it requires that users edit the BIOS settings to allow for VMWare to be executed.

AlignTool is designed for academic, i.e., non-profit, research purposes only and is made available to members of academic institutions only. It makes use of the WebMAUS web service for the automatic alignment of spoken utterances and their literal orthographic transcriptions under the Conditions of Use for Academic Institutions as specified by the Bavarian Archive for Speech Signals at the Ludwig-Maximilians-University Munich (BAS) (see BAS, 2017c). In addition, users need MS Excel or equivalents (OpenOffice/ LibreOffice) and Praat to use AlignTool. Within the virtual environment, AlignTool and WebMAUS are both easy to use. AlignTool and its documentation are available at https://www.linguistics.ruhr-uni-bochum.de/~belke/aligntool.shtml.

## Basic concepts and processing steps

The function of this section is to provide an overview of AlignTool's functionality. Users of AlignTool are referred to the AlignTool User Manual (Belke, Keite, & Schillingmann, 2017; see link provided above) to find out more about how to use AlignTool for analyzing their own data. AlignTool is geared towards analyzing files from three different types of recordings as illustrated schematically in the top row of Fig. 1. The first two types pertain to experimental settings consisting of sequences of trials. Each trial features a stimulus that the participant is asked to react to, for instance, an object or a word that the participant is asked to name. In *recordings of multiple trials* in one audio file, the trial sequence of a full experiment or experimental block is recorded. Whenever a new stimulus is shown, audio signals, e.g., beeps, are emitted and are recorded alongside the participant's responses so that the temporal relation between stimulus onset and the participant's response can be analyzed off-line. In *trial-by-trial recordings*, a new audio file is generated on every trial, recording the participant's response. Recordings are typically made from the onset of the stimulus until some time after the participant has responded. In *recordings of semi-spontaneous speech*, participants are recorded in an experimental setting but are less constrained in terms of the content and the length of the utterances they generate. We will focus on the first two types of recordings in the beginning of this section and return to recordings of semi-spontaneous speech at the end.

When using AlignTool, users work with and coordinate three groups of files: wav audio files, TextGrid files, and an Excel file (the workbook) that includes (a) lists of commands to be executed by AlignTool (sheet "batch"; see Fig. 2), (b) a copy of the intervals and their contents stored in the TextGrid files (sheet "segments"; see Fig. 3), and, eventually, (c) the onsets and offsets of all words analyzed by AlignTool (sheet "on_offsets").

Figure 1 gives an overview of the workflow for each type of recording listed above. In the following sections, we briefly describe the main steps in the workflow.

**Prepare Excel workbook and working directory** In a first step, users need to prepare a working directory, copying all wav files that are to be analyzed into a folder "wav" in this directory. Next, they prepare the "batch" sheet of the Excel workbook (Fig. 2). To this end, they first make AlignTool retrieve the wav files and write their names into the workbook. AlignTool simultaneously specifies the corresponding TextGrid file names. After that, users set the parameters of the commands to be executed in the following steps.

**Initial segmentation (where applicable) and creation of *seg.beep* tier (*segmentBeeps*)** In the second step, the TextGrid files are created and a new tier called *seg.beep* is added (see Fig. 4). Its function is to segment the recording into trials, as required for recordings of multiple trials in one audio file. In these recordings, users will typically have recorded the participant's utterances in one channel and an audio signal indicating the onset of a trial in the other channel, as illustrated in Fig. 1 (top left). AlignTool uses Praat to automatically detect the trial onsets and to create trial-by-trial intervals. It labels all intervals identified as beeps "beep" in the *seg.beep* tier and all periods between two beeps and after the last beep as "speech" (see Fig. 4).

Strictly speaking, a segmentation into trials is not required for trial-by-trial recordings or recordings of semi-spontaneous speech, as these do not include trial onset beeps. However, as subsequent processing stages in AlignTool require a tier called *seg.beep*, we recommend that users generate TextGrid files with a *seg.beep* tier nonetheless. They can do so using the command *addTier* (see Fig. 1). We detail how to do this in the User Manual (Belke et al., 2017).

When a new tier has been created in the TextGrid files, users can import the information into the "segments" sheet of the Excel workbook using the command *Import from TextGrids*. Each interval represented in a tier in the TextGrid file will then be represented in one line of the "segments" sheet. Lines are listed in a tier-by-tier fashion for each file,

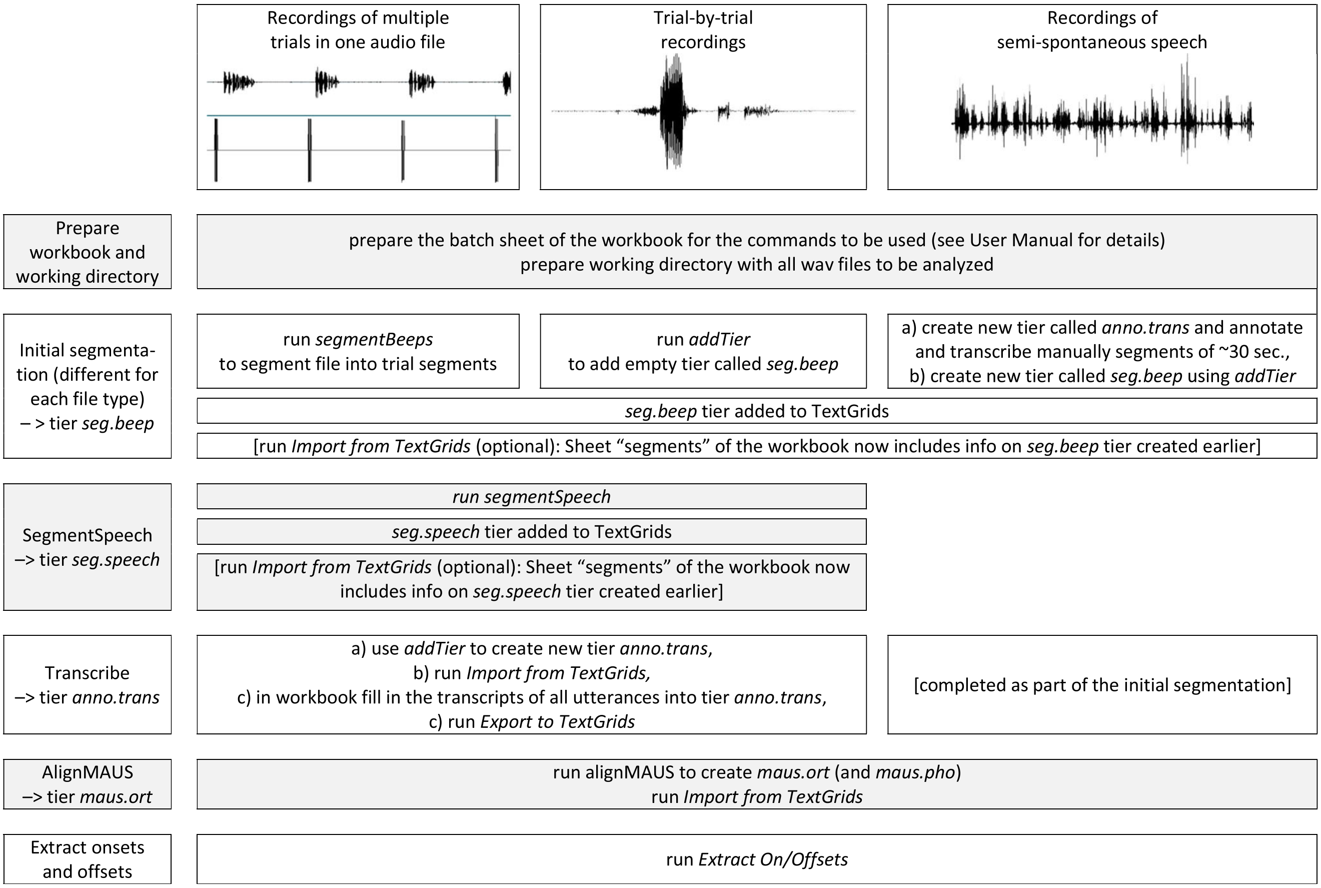
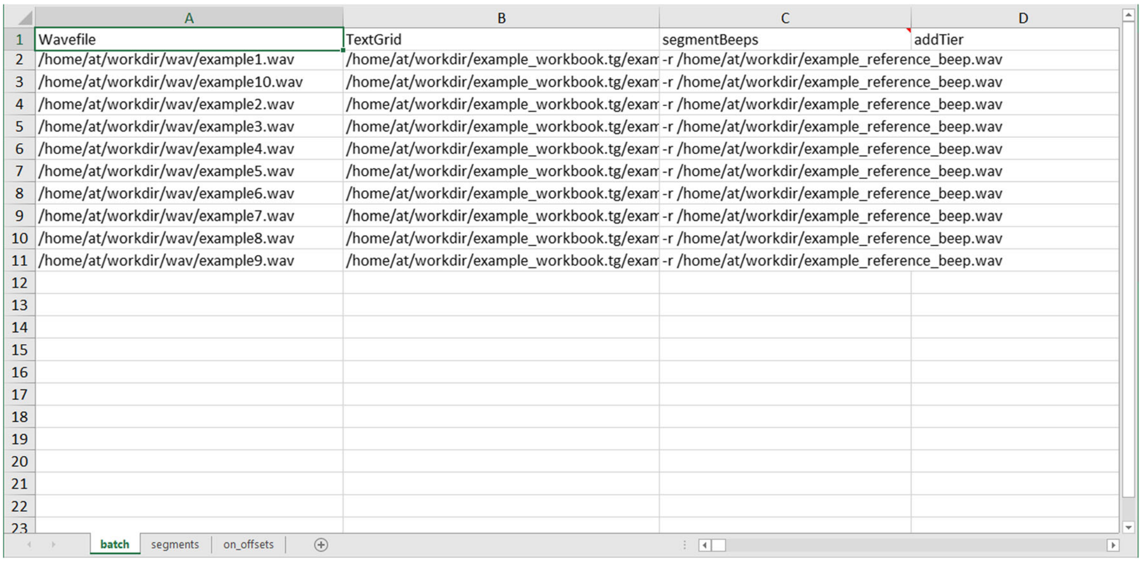

Recordings of multiple trials in one audio file
Trial-by-trial recordings
Recordings of semi-spontaneous speech
Prepare workbook and working directory
prepare the batch sheet of the workbook for the commands to be used (see User Manual for details)
prepare working directory with all wav files to be analyzed
Initial segmentation (different for each file type) – > tier *seg.beep*
run *segmentBeeps* to segment file into trial segments
run *addTier* to add empty tier called *seg.beep*
a) create new tier called *anno.trans* and annotate and transcribe manually segments of ~30 sec.,
b) create new tier called *seg.beep* using *addTier*
*seg.beep* tier added to TextGrids
[run *Import from TextGrids* (optional): Sheet "segments" of the workbook now includes info on *seg.beep* tier created earlier]
SegmentSpeech –> tier *seg.speech*
*run segmentSpeech*
*seg.speech* tier added to TextGrids
[run *Import from TextGrids* (optional): Sheet "segments" of the workbook now includes info on *seg.speech* tier created earlier]
Transcribe –> tier *anno.trans*
a) use *addTier* to create new tier *anno.trans*,
b) run *Import from TextGrids,*
c) in workbook fill in the transcripts of all utterances into tier *anno.trans*,
c) run *Export to TextGrids*
[completed as part of the initial segmentation]
AlignMAUS –> tier *maus.ort*
run alignMAUS to create *maus.ort* (and *maus.pho*)
run *Import from TextGrids*
Extract onsets and offsets
run *Extract On/Offsets*

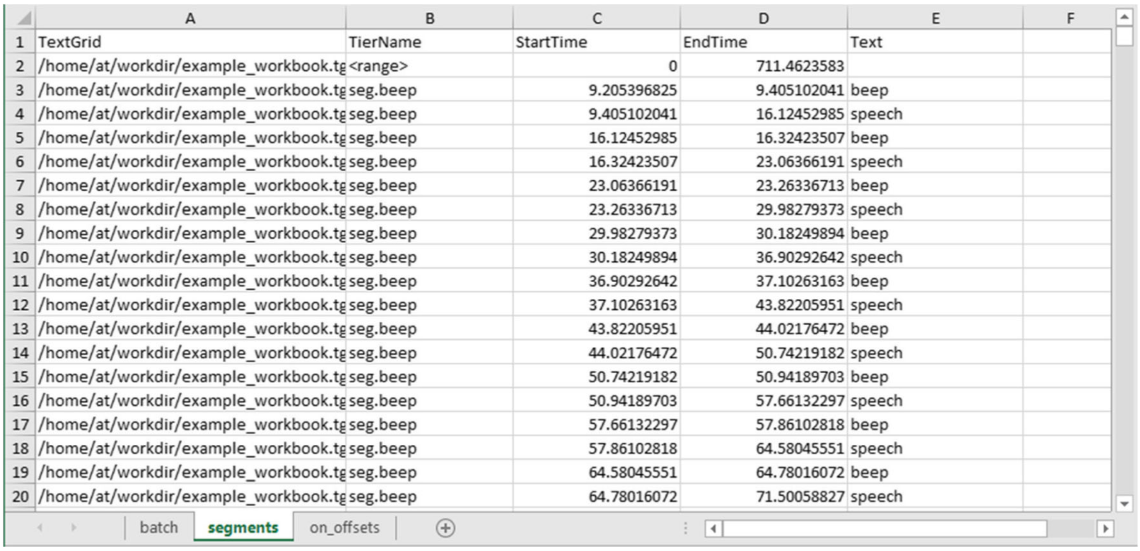
**Fig. 1** Overview of the processing steps required to align automatically the three types of audio recordings

with all lines pertaining to the first tier being listed first, all lines pertaining to the second tier second, and so forth. Figure 3 presents the "segments" sheet for the *seg.beep* tier of a recording of multiple trials. It displays the "beep" and "speech" segments and their onset and offset times as established during the *segmentBeeps* procedure.



| | A | B | C | D |
|---|---|---|---|---|
| 1 | Wavefile | TextGrid | segmentBeeps | addTier |
| 2 | /home/at/workdir/wav/example1.wav | /home/at/workdir/example_workbook.tg/exam | -r /home/at/workdir/example_reference_beep.wav | |
| 3 | /home/at/workdir/wav/example10.wav | /home/at/workdir/example_workbook.tg/exam | -r /home/at/workdir/example_reference_beep.wav | |
| 4 | /home/at/workdir/wav/example2.wav | /home/at/workdir/example_workbook.tg/exam | -r /home/at/workdir/example_reference_beep.wav | |
| 5 | /home/at/workdir/wav/example3.wav | /home/at/workdir/example_workbook.tg/exam | -r /home/at/workdir/example_reference_beep.wav | |
| 6 | /home/at/workdir/wav/example4.wav | /home/at/workdir/example_workbook.tg/exam | -r /home/at/workdir/example_reference_beep.wav | |
| 7 | /home/at/workdir/wav/example5.wav | /home/at/workdir/example_workbook.tg/exam | -r /home/at/workdir/example_reference_beep.wav | |
| 8 | /home/at/workdir/wav/example6.wav | /home/at/workdir/example_workbook.tg/exam | -r /home/at/workdir/example_reference_beep.wav | |
| 9 | /home/at/workdir/wav/example7.wav | /home/at/workdir/example_workbook.tg/exam | -r /home/at/workdir/example_reference_beep.wav | |
| 10 | /home/at/workdir/wav/example8.wav | /home/at/workdir/example_workbook.tg/exam | -r /home/at/workdir/example_reference_beep.wav | |
| 11 | /home/at/workdir/wav/example9.wav | /home/at/workdir/example_workbook.tg/exam | -r /home/at/workdir/example_reference_beep.wav | |
| 12 | | | | |
| 13 | | | | |
| 14 | | | | |
| 15 | | | | |
| 16 | | | | |
| 17 | | | | |
| 18 | | | | |
| 19 | | | | |
| 20 | | | | |
| 21 | | | | |
| 22 | | | | |
| 23 | | | | |

**Fig. 2** Example of the "batch" sheet for 10 audio files. AlignTool works through them line by line as soon as the user initiates a command (e.g., *segmentBeeps*). All wav and TextGrid files are specified in the first two columns and the parameters for using the command *segmentBeeps* are set. The parameters for using the other commands have yet to be added



| | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| 1 | TextGrid | TierName | StartTime | EndTime | Text | |
| 2 | /home/at/workdir/example_workbook.tg | <range> | 0 | 711.4623583 | | |
| 3 | /home/at/workdir/example_workbook.tg | seg.beep | 9.205396825 | 9.405102041 | beep | |
| 4 | /home/at/workdir/example_workbook.tg | seg.beep | 9.405102041 | 16.12452985 | speech | |
| 5 | /home/at/workdir/example_workbook.tg | seg.beep | 16.12452985 | 16.32423507 | beep | |
| 6 | /home/at/workdir/example_workbook.tg | seg.beep | 16.32423507 | 23.06366191 | speech | |
| 7 | /home/at/workdir/example_workbook.tg | seg.beep | 23.06366191 | 23.26336713 | beep | |
| 8 | /home/at/workdir/example_workbook.tg | seg.beep | 23.26336713 | 29.98279373 | speech | |
| 9 | /home/at/workdir/example_workbook.tg | seg.beep | 29.98279373 | 30.18249894 | beep | |
| 10 | /home/at/workdir/example_workbook.tg | seg.beep | 30.18249894 | 36.90292642 | speech | |
| 11 | /home/at/workdir/example_workbook.tg | seg.beep | 36.90292642 | 37.10263163 | beep | |
| 12 | /home/at/workdir/example_workbook.tg | seg.beep | 37.10263163 | 43.82205951 | speech | |
| 13 | /home/at/workdir/example_workbook.tg | seg.beep | 43.82205951 | 44.02176472 | beep | |
| 14 | /home/at/workdir/example_workbook.tg | seg.beep | 44.02176472 | 50.74219182 | speech | |
| 15 | /home/at/workdir/example_workbook.tg | seg.beep | 50.74219182 | 50.94189703 | beep | |
| 16 | /home/at/workdir/example_workbook.tg | seg.beep | 50.94189703 | 57.66132297 | speech | |
| 17 | /home/at/workdir/example_workbook.tg | seg.beep | 57.66132297 | 57.86102818 | beep | |
| 18 | /home/at/workdir/example_workbook.tg | seg.beep | 57.86102818 | 64.58045551 | speech | |
| 19 | /home/at/workdir/example_workbook.tg | seg.beep | 64.58045551 | 64.78016072 | beep | |
| 20 | /home/at/workdir/example_workbook.tg | seg.beep | 64.78016072 | 71.50058827 | speech | |

Fig. 3 Example of the "segments" sheet

**Detecting speech in trials using Praat (*segmentSpeech*)** *segmentSpeech* searches for relevant speech events based on the intervals established in the *seg.beep* tier (see Fig. 1). For instance, in beep-segmented recordings of multiple trials, *segmentSpeech* will search each interval labelled "speech" in the *seg.beep* tier to find the actual speech onset and offset. In trial-by-trial recordings featuring one trial per file, *segmentSpeech* will search each file for the onset and offset of the speech signal. Given that AlignTool's primary function is to obtain word onset and offset times of one utterance per trial, *segmentSpeech* assumes that one speech chunk is present per trial and searches for the beginning and the end of this chunk. Pauses within the utterance are ignored by *segmentSpeech*. Note though that they are detected in a later processing step, *alignMAUS* (see below), provided that they are not too long. *segmentSpeech* creates a new tier called *seg.speech*, labelling the intervals of speech it detected as "speech" (see Fig. 4). Users can synchronize the new information in the TextGrid files with the Excel workbook by means of *Import from TextGrids* function.

**Creating literal transcriptions** In the next processing step, users need to create literal transcriptions of the speech included in the "speech" intervals identified previously in the

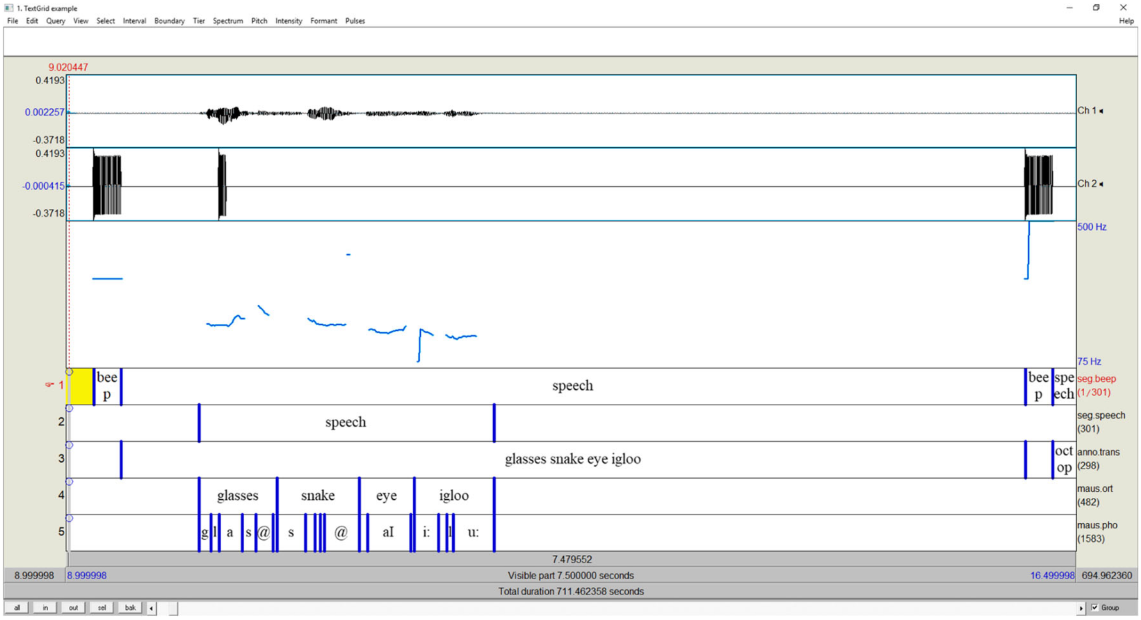


Fig. 4 Example of a fully annotated beep-segmented recording of multiple trials. Trial-onset beeps and speech are recorded on separate channels (see top half of panel). The speech signal features an average signal-to-noise ratio. The bottom half of the screenshot presents the tiers generated by AlignTool during the course of the automatic annotation

*seg.speech* tier. To this end, they first use the function *addTier* to add a new tier called *anno.trans* to the TextGrid files, specifying as preliminary text to be entered in all intervals "TODO". Next, they import the contents of the *anno.trans* tier to the Excel workbook, where they can replace all "TODO"-entries by the transcriptions of the participants' utterances. We provide some recommendations about how to best do this in the User Manual (Belke et al., 2017), as users will often be able to paste in the transcripts from the individual trial descriptions for the participant, which typically include the target utterances. Once all transcriptions have been entered into the Excel workbook, users can transfer the transcriptions from the Excel workbook to the TextGrid file by means of the *Export to TextGrids* function of AlignTool.

**Aligning the speech signal with transcriptions using WebMAUS** In this processing step, WebMAUS (Kisler et al., 2016), the web service providing forced alignments using MAUS (Schiel, 1999), is used to generate forced alignments of the speech intervals in the audio signal and the transcription the user has provided for these intervals. This processing step creates two new tiers – a tier representing the onset and offset times of words (*maus.ort*) and a tier representing onset and offset times of individual phonemes (*maus.pho*, see Fig. 4). Using *Import from TextGrids*, these tiers can be imported to the Excel workbook.

As stated above, AlignTool uses the WebMAUS service, which is based on MAUS (BAS, 2017a; Schiel, 1999, 2015), to carry out this processing step. MAUS is an automatic speech recognition system based on continuous HMMs (Hidden Markov Models; see also Appendix 1). Unlike most other automatic speech recognition systems, MAUS is geared explicitly towards performing forced alignments. AlignTool uses MAUS in conjunction with the BAS Grapheme-Phoneme Converter (G2P; see BAS, 2017b) service to generate automatic phonetic transcriptions of the orthographic transcriptions. This service is provided for a variety of languages, including German, Dutch, and British English, which we focused on in our evaluation of AlignTool's accuracy, but also French, Italian, Spanish, Finnish, and Russian (for a full list, see the "service options" menu on the WebMAUS web page, https://clarin.phonetik.uni-muenchen.de/BASWebServices/interface/WebMAUSBasic. This allows for AlignTool to be applied to more than just the three languages we report on. It also allows for AlignTool to be used on pseudoword utterances, as long as the pseudowords are phonotactically plausible in the speakers' language, so that G2P can generate a pronunciation based on the pseudowords' orthography.

Users of AlignTool who use *alignMAUS* accept the "Conditions of Use" of the Bavarian Archive for Speech Signals Webservices (cf. BAS, 2017c). We therefore encourage all users to read these conditions before using *alignMAUS*.

**Extract On/Offsets** When all processing steps for analyzing the audio files have been completed, the "segments" sheet of the Excel workbook contains information on all tiers in the TextGrid file, i.e., onset and offset times of the intervals corresponding to individual trials, words in trials and phonemes within words. *Extract On/Offsets* adds a new sheet to the workbook ("on_offsets") and computes the onset and offset times of each of these events in relation to the trial onset. In recordings of multiple trials, this will be the onset of the beep; in recordings of individual trials or semi-spontaneous speech, this will be the onset of the recording.

**Aligning semi-spontaneous speech** When aligning semi-spontaneous speech, users should pre-segment the audio files into intervals of about 30 s or less (the most suitable interval length may vary depending on the properties of the spoken utterances (number of pauses etc.) and the quality of the audio recording). To this end, users need to create a TextGrid file for each recording with a tier called *seg.beep* and a tier called *anno.trans*. Starting with the tier *anno.trans*, users segment the audio file and enter the literal transcriptions of the utterances included in each interval into the *anno.trans* tier. Next, they use the *addTier* function to copy the *anno.trans* tier to a tier called *seg.beep*, including the label "speech" in each of the intervals specified previously in the tier *anno.trans*. As of here, *alignMAUS* can be applied and the procedure is the same as described above.

**Inspection and manual corrections of AlignTool results and parameter optimization** As AlignTool does not stop processing when an error occurs, users need to check the log file for potential processing errors. The User Manual (Belke et al., 2017) includes suggestions about how to do this. We also recommend that users open the TextGrid and audio files for one (or more) audio files repeatedly during the course of the analysis to check whether AlignTool has made the intended changes to the TextGrid files correctly and whether the measurements are accurate. At the very latest, users should inspect the temporal annotation provided by AlignTool after carrying out the forced alignments using *AlignMAUS*.

Users can contribute to yielding optimal preliminary temporal annotations from AlignTool by optimizing the parameters for each processing step. This applies primarily to *segmentBeeps* and *segmentSpeech*, but also to *alignMAUS*. For instance, in order to optimize the accuracy of *segmentSpeech*, users can specify how much (in db) the audio signal must differ from the silence threshold in order to be classified as speech. This parameter should be used with care though, as a too conservative (i.e., too high) setting may cause AlignTool not to detect voiceless fricatives or other speech events that differ little from the silence threshold.

A full list of parameters and their functions is provided in the User Manual (see Appendix A in Belke et al., 2017).

Which parameters are suited best for which recording depends to a large extent on the quality of the recording and the kind of background noise, so we cannot give any general recommendations on the parameter settings. However, in Appendix A6 of the User Manual we give recommendations on how users can find optimal parameters for their recordings (Belke et al., 2017). Table 1 presents an overview of the parameter settings we used for the evaluation of AlignTool, detailed in the next section. They can serve as orientation for the parameter settings users may want to use for their own analyses.

We recommend that users annotate the recordings of a few participants manually so as to be able to evaluate the quality of the alignment under their parameter settings in AlignTool with respect to this gold standard. In our experience, an optimal parameter setting for a given recording scenario can be carried forth to new recordings from the same scenario. Therefore, it is worthwhile to invest some tuning effort when first using AlignTool.

## Evaluation of AlignTool

Manually established measurements of onset and offset times of words in participant recordings are the gold standard for evaluating tools like AlignTool. We compiled a sample of participant recordings obtained in experimental and semi-experimental settings typical of language production research and annotated them manually. We used them to evaluate the accuracy of the measurements generated automatically by AlignTool. Note that the tool is designed in such a way that users can easily inspect its accuracy and correct the measurements, where necessary. Hence, its actual accuracy in proper use will be much higher. The evaluations reported below can serve as an estimate of how exact a purely automatic annotation with AlignTool can be.

### Speech corpora I: single and multiple word utterances

We used two corpora to assess AlignTool's accuracy in establishing the temporal onsets and offsets of words in spoken utterances of one to four words recorded in an experimental setting. By assessing the accuracy of determining the temporal onset of the first word in an utterance, we specifically assessed AlignTool's accuracy as a voice-key. In doing so, we also compared its performance to the performance of a custom-made hardware voice-key (Hasomed NesuBox 2) and of SayWhen and Chronset, the software-based voice-keys presented by Jansen and Watter (2008) and Roux et al. (2016), respectively, both of which we could apply to our data.

**Rastle&Davis corpus (Rastle & Davis, 2002; English)** As outlined in the Introduction, Rastle and Davis (2002) had 24 participants name two groups of 20 words, one beginning with /s/ (simple onset) and one beginning with /sp/ or /st/ (complex onset). The speakers were participants from the University of Cambridge (cf. Rastle & Davis, 2002, p. 309). Data from two participants had to be excluded due to technical problems, yielding a total of 880 critical trials. An additional set of filler words, none of which started with /s/, were recorded but were not considered in the analyses. Onset times were measured manually (for details, see Rastle & Davis, 2002) and using two different types of voice-key. For the purpose of this evaluation, we shall focus on the manual measurements of participants' speech onset times, which were longer for simple than for complex onsets.

The parameters for analyzing the word onset and offset times in the Rastle&Davis corpus with AlignTool are presented in Table 1. In order to measure the onset times of the utterances with SayWhen, we pre-processed the data so as to bring them into a format suitable for SayWhen. First, we resampled the recordings from 22,050 Hz to 44,100 Hz, using SoX (sox [inputfile].wav -r 44100 [outputfile.wav]). Next, we concatenated the trialwise recordings of the wav files to one wav file in order to simulate a recording of a full experiment. In order for this file to be processed by SayWhen, it needed to also include a 10-ms trial onset signal, which had not been part of the original recordings, on the left channel. We incorporated this marker in the concatenation process by including a trial onset signal file (with the trial onset signal on the left channel and silence on the right channel) before each trial recording. In addition, we added 30 ms of silence at the beginning and the end of the concatenated files, as this was necessary for SayWhen to find the first and last trials reliably. The concatenated file thus included 30 ms silence at the beginning, followed by a series of pairs of trial onset signal files and trial recordings, and 30 ms silence at the end. A new wav file header was added to the concatenated audio file and it was entered to SayWhen (using default settings). The onset latencies established by SayWhen were saved to a CSV file. In a last processing step, we subtracted 10 ms from all latencies provided by SayWhen, as its measurements were started at the beginning of the 10-ms trial onset marker, which had not been part of the original trial recording.

**MTAS corpus (English, German, Dutch)** The MTAS (short for Manually Temporally Annotated Speech) corpus, was specifically created for the evaluation of AlignTool: We collected data from 30 native speakers of German, 30 native speakers of Dutch, and 30 native speakers of British English, who all completed an object-naming and a word-reading task. We chose these two tasks as they are frequently used in language production research. The participants were recruited via the participant pools of the Department of Linguistics at Ruhr-University Bochum, the Max Planck Institute for Psycholinguistics in Nijmegen, and the School of Psychology at the University of Birmingham, respectively. We did not record the region of origin of the speakers and their accents.

**Table 1** Size and recording quality of the corpora used for evaluating AlignTool and analysis parameters used with AlignTool for each corpus

| | Defaults | Single- and multiple-word utterances | | | | Semi-spontaneous speech | | |
|---|---|---|---|---|---|---|---|---|
| | | R&D | MTAS German | MTAS Dutch | MTAS English | German Display Comparison | Dutch Meyer & Sjerps | English Map Task |
| Quality of Recording | | Excellent | Average | Excellent | Poor | Poor | Excellent | Excellent |
| No. of speakers (total / used) | | 22 / 21 | 30 / 30 | 30 / 28 | 30 / 30 | 7 / 7 | 36 / 36 | 21 / 21 |
| No. of words (total / used) | | 880 / 840 | 9,000 / 8,367 | 9000 / 7,081 | 9,000 / 7,876 | 10,822 / 8,955 | 12,637 / 3,972 | 13,619 / 12,190 |
| segmentBeeps | | | | | | | | |
| --refbeep <refbeep> | none | n/a | refbeep_D.wav | refbeep_NL.wav | refbeep_E.wav | n/a | n/a | n/a |
| --mincorrelation <0-1> | 0 | | 0 | 1 | 0 | | | |
| --silencethreshold <db> | -25 | | -25 | -25 | -25 | | | |
| --minsounding <s> | .180 | | .180 | .180 | .180 | | | |
| --seekflank | false | | true | true | true | | | |
| segmentSpeech | | | | | | | | |
| --filter-tier <tier> | none | seg.beep[a] | seg.beep | seg.beep | seg.beep | n/a | seg.beep[a] | n/a |
| --denoise | false | false | false | false | false | | false | |
| --shiftonsets <s> | 0 | 0 | 0 | 0 | 0.02 | | 0 | |
| --shiftoffsets <s> | 0 | 0 | 0 | 0 | 0.02 | | 0 | |
| --trainwindow <s> | 1 | 0.1 | 1 | 1 | 1 | | 1 | |
| --speechthresh <f> | 0.5 | 0.4 | 0.2 | 0.2 | 0.3 | | 0.0 | |
| --snradd <db> | 1 | 9 | 2 | 2 | 8 | | 18 | |
| alignMAUS | | | | | | | | |
| --language <language> | deu-DE | eng-GB | deu-DE | nld-NL | eng-GB | deu-DE | nld-NL | eng-GB |
| --filter-tier <tier> | seg.beep | seg.speech | seg.beep | seg.beep | seg.beep | seg.speech[b] | seg.beep | seg.speech[b] |
| --segmentation-tier <tier> | seg.beep | seg.speech | seg.speech | seg.speech | seg.speech | seg.speech[b] | seg.speech | seg.speech[b] |
| --denoise | false | false | false | false | false | false | false | false |
| --initialsilence | false | false | true | true | false | false | false | true |
| --remote | false | false | false | false | false | false | false | false |

[a] As there were no beeps for segmenting trials in these corpora, a functional equivalent of the *seg.beep* tier was created using the *addTier* command

[b] *seg.speech* was based on the onset of the first and the offset of the last word per recording, as established during the manual temporal analysis

For the single object-naming task, we selected 100 pictures of objects for each language from the databases provided by Bates et al. (2003) for German and English and by Severens, van Lommel, Ratinckx, and Hartsuiker (2005) for Dutch (see Appendix 2). These databases include details on the frequency of the object names in the respective languages (cf. Baayen, Piepenbrook, & van Rijn, 1995) and the name agreement associated with the pictures of the objects – the more likely the participants are to use the same word(s) for a given object, the higher its name agreement. In selecting the objects, we made sure that their names were frequent words and that the pictures had high name agreement, meaning that in the norming studies, pictures were associated with three different names at most. In addition, we selected the stimuli in such a way that their names covered a variety of different onset phonemes. Occasionally, this required applying a less strict criterion on name agreement. The full list of stimuli is provided in Appendix 2. Participants were asked to name the pictures as fluently and accurately as possible.

In the word-reading task, the same object names were used as in the object-naming task, but they were combined into a total of 50 different four-word combinations, such as *tent dog grapes skeleton*. Each object name featured twice across all word combinations. The word combinations were compiled to include words that ended and began with the same phoneme (*grapes – skeleton*), were very similar (*frog – clock*) or very dissimilar phonemes (*broom – chair*), so as to model easy and hard conditions for establishing precisely the onset and offset times of individual words within the utterance. Again, participants were asked to read the object names out loud as fluently and accurately as possible. During testing, we used a custom-made hardware voice-key (Hasomed NesuBox 2) for establishing participants' utterance onset times online, allowing us to evaluate AlignTool against this voice-key. The voice-key emitted a beep signal when it was triggered, allowing the experimenter to record all trials when the voice-key was triggered audibly too early, too late, or not at all.

Regarding the manner of articulation, the sets included a large number of picture names with fricative onsets (34 for German, 33 for English, and 39 for Dutch, respectively) and plosive onsets (39 for German, 35 for English, and 29 for Dutch). Vowels and semivowels were less frequent onsets (17 for German, 20 for English, and 21 for Dutch) and nasals approximants and trills were rather rare. This distribution reflects that the frequency of occurrence of phonemes in the onsets of words varies. Some onsets are not included in the materials because they either did not constitute word onsets in any of the three languages or because the name agreement of the words containing the onsets was too low.

For the word-reading task, we assembled the written names of the objects selected for the single object naming to 50 lists of four names. Within each list, we manipulated the similarity of the final phoneme of the first (second, third) word and the first phoneme of the second (third, fourth) word. Below, we refer to these pairs of offsets and onsets of consecutive words as transitions. With 50 four-word lists per language and three transitions per list, there were 150 transitions per language. Across word groups, we identified five categories of transitions between words, depending on the similarity between the two phonemes at the transition between two words:

– The two phonemes differed in both place and manner of articulation.
– The two phonemes were similar, i.e., they shared the same manner of articulation but differed in the place of articulation (as in *tent – plate*) or vice versa (as in *plate – swan*), or they shared the same place and manner of articulation, but differed in voicing (as in *frog – clock*).
– The two phonemes were identical (as in *grapes – skeleton*).

The data collection procedure was identical across the three languages. All participants were tested in both the single-object-naming and the multiple-word-reading tasks in their respective native language. We asked them to complete the multiple-word-naming task first, as we hoped that familiarizing the participants with the object names in this task would increase name agreement in the single-object-naming task. Participants received written instructions prior to each task and were given the opportunity to ask questions. Each task was preceded by three practice trials so as to familiarize the participants with the task and the procedure. In the word-naming task, a fixation point was shown for 500 ms, followed by the four words for 6 s and a blank screen for 150 ms. After that, the next trial was initiated. In the single object-naming task, the same trial timing was used, but the object was presented for 2 s and the blank screen between two trials was shown for 750 ms. The apparatus was also parallel across the three languages, featuring standard desktop Pentium computers for controlling the stimulus presentation and 17-in. to 19-in. computer screens for presenting the stimuli using the NESU software (Nijmegen Experiment SetUp).

In the English and Dutch testing settings, participants were seated in a quiet room; the German recordings were made with participants seated in a sound proof booth. The responses of the participants were registered using a Sony ECM-MS907 microphone (German and English) and a Sennheiser ME64 microphone (Dutch), respectively. The signal was fed through an external voice-key (Hasomed NesuBox 2) on to a second computer for recording (German) and an external DAT recorder (Dutch), respectively. For the English recordings, we had planned to use the same setup for DAT recordings as in Dutch. However, due to technical difficulties, we had to record the utterances with an M-Audio MicroTrack II recorder, which produced recordings of very poor quality. In the end, the recording quality was excellent for the Dutch utterances, average for the German and very poor for the English data. These unintended differences in recording quality allow us to evaluate how much the three software-based analysis tools, AlignTool, SayWhen, and Chronset, are affected by differences in recording quality.

The parameters for analyzing the word onset and offset times in the MTAS corpus with AlignTool are presented in Table 1. In order to measure the onset times of the utterances with SayWhen, we first split our audio recordings of the experimental sessions into trialwise recordings, based on the trial onset beeps we had recorded during the experiment. These trialwise recordings were treated in the same way as the recordings of the Rastle&Davis corpus, with the exception that no resampling was necessary.

## Speech corpora II: semi-spontaneous speech

The semi-spontaneous speech data were used to pilot AlignTool's accuracy in analyzing the temporal structure of semi-spontaneous speech elicited in description tasks.

**Display Comparison Task corpus (Sichelschmidt et al., 2010; German)** For German, we used a subset of a corpus of utterances recorded from pairs of speakers engaged in semi-spontaneous dialogues (see Sichelschmidt, Jang, Kösling, Ritter, &Weiß, 2010). Each of them saw a set of colored objects on the screen but was unable to see the display of their partner. The displays differed in only one detail and the dialogue partners' task was to describe their displays to each other so as to identify the difference. Speakers were recruited at Bielefeld University (Sichelschmidt et al., 2010). We have no detailed information on the speaker characteristics. In total, we annotated 10,822 words from seven pairs of speakers. The signal-to-noise ratio of the recordings was poor, as they unavoidably included noise generated by the computers and other background noise in the room. The parameters for analyzing the word onset and offset times in the Display Comparison Task corpus with AlignTool are presented in Table 1.

**Sjerps&Meyer corpus (Sjerps & Meyer, 2015; Dutch)** For the evaluation in Dutch, we employed data from an experiment using a pseudo-dialogue setting (Sjerps & Meyer, 2015; Experiment 1, Speaking Only task). Participants described the spatial positioning of two pairs of objects, using sentences of the form "put the A above (below) the B and put the C below (above) the D". Manual annotations of the onset and offset times were available for all nouns in the utterances (A, B, C, and D). Participants were native speakers of Dutch and were recruited from the participant pool of the Max Planck Institute for Psycholinguistics in Nijmegen. We have no detailed information on the speaker characteristics.

We selected correct responses only, yielding a total of 993 utterances with 12,367 words, 3,972 of which had been annotated manually. Originally, we had planned to include a second set of utterances from the Tapping and Speaking task, which required participants to tap rhythmically while speaking. Unfortunately, the tapping noise was clearly audible in the recordings and made it impossible for AlignTool to operate reliably, so we could not include these data. The parameters for analyzing the word onset and offset times in the Sjerps&Meyer corpus with AlignTool are presented in Table 1.

**Map Task corpus (Anderson et al., 1991; English)** This corpus includes route descriptions of 64 different speakers, most of whom were Scottish and were from "within a 30 mile radius of the center of Glasgow" (Anderson et al., 1991, p. 361). The recordings include the speech of an instructor and a dialogue partner recorded on separate channels. We converted the .ses (raw) audio files to wav files using the Linux-based SoX utility tool and transcribed and temporally annotated a total of 13,619 words of the instructor in the recordings of 21 pairs of speakers. We selected those recordings because they included only few intervals where the two speakers spoke simultaneously. The parameters for analyzing the word onset and offset times in the Map Task corpus with AlignTool are presented in Table 1.

## Manual annotations of word onsets and offsets

We used the AVS audio editor version 7.2.1.487 to annotate the words manually. With this tool, it is possible to set markers within an audio file and create a "marker list" in which markers can be added, merged, renamed, replayed, and saved into an xml file. The values stored in the xml file provide the time stamp of a marker, multiplied by the sampling rate. The annotation rules are summarized in Appendix 3.

After 2–3 weeks into the annotation process we double-checked the marked onsets and offsets by exchanging the annotated data among the annotators and controlling whether they would all have annotated the data as their colleagues had done. This was usually the case. After this, we addressed problems that often occurred. For instance, it turned out that word-final plosives like /k/ and /t/ had sometimes been left unmarked. We corrected the data accordingly and also checked whether the first annotator had observed the rule that successive words should always be 1 ms apart. In a second round of corrections, we distributed the data in such a way that they were assigned to annotators who had not previously seen them and had each annotator check 15 % of the data as to whether he or she would have marked the same word beginnings and endings. Whenever annotators differed by 10 ms or more, they were asked to correct the marker and if it became clear that in a file the accordance was off multiple times, they were asked to check the entire file.

At the end of the annotation process, each file was assigned to yet another annotator who annotated about 10 % of the data from scratch (see Table 2). These annotations were used to assess the consistency of the annotations across annotators. For the sake of comparability, we restricted the consistency analyses to those data points that were also included in the evaluation of AlignTool (see below). Table 2 presents the average differences between annotators in ms (averaging across differences greater and smaller than 0), as well as the average absolute difference between annotators. It also provides the standard deviation of the measurements provided by each annotator, the covariance between annotators and their correlation (Pearson's *r*). In addition, we provide intraclass correlation (ICC) scores (ICC(C,1); McGraw & Wong, 1996) as a measure of consistency. The differences were small and the ICCs exceptionally high throughout. However, they display a small dip for the English MTAS data, which is most likely due to the poor audio quality of these data.

## Comparison of automatic and manual annotations: single and multiple word utterances

**Rastle&Davis corpus (Rastle & Davis, 2002; English)** One participant was excluded from the analyses as this person inhaled audibly on almost all trials, which caused substantial deviations of the automatic temporal alignments provided by AlignTool from the manual measurements. Such deviations would usually be corrected manually but as the present evaluation was geared towards establishing the results of the automatic temporal alignments alone, we excluded this participant from the analyses. Of the remaining data, 18 trials were excluded due to participant errors. Table 3 lists the mean utterance onset times established manually and using AlignTool, SayWhen, and Chronset. The results of the manual annotations are given for all 22 participants originally included in the analyses reported by Rastle and Davis (2002) and for the subset of 21 participants included in the present analysis. Rastle and Davis had found that with their manual annotation, response times were 9 ms faster for complex than for simple onsets. This effect was significant by participants and approached significance by items ($p = .05$). Excluding one participant yielded an effect of 8 ms with $p < .05$ for the by-participants and $p = .061$ for the by-items analysis.

AlignTool automatically annotated the onset times in the two conditions about 33 ms earlier than the manual annotations. The reduction in response time was slightly more pronounced in the simple than in the complex condition, reducing the effect of condition to 2 ms (n.s.). Critically, the reduction

**Table 2** Average differences (in ms, with standard deviations) between annotators, averaging across differences greater and smaller than 0, average absolute differences (in ms, with standard deviations) and measures of variability per annotator and across annotators (standard deviations (in ms), Person's r and ICC). For comparability, analyses were restricted to the data points included in the analyses for evaluating AlignTool's accuracy

| | Mean difference | | Mean absolute difference | | *SD* annotator 1 | *SD* annotator 2 | Covariance | Pearson's *r* | ICC (95 % CI) |
|---|---|---|---|---|---|---|---|---|---|
| MTAS Corpus, Onsets | | | | | | | | | |
| German[a] | 5 | (29) | 17 | (24) | 600 | 596 | 357,134 | .999 | .999 (.001) |
| Dutch[b] | 2 | (30) | 18 | (25) | 647 | 651 | 420,535 | .999 | .999 (.001) |
| English[c] | 26 | (312) | 54 | (309) | 610 | 693 | 377,527 | .893 | .885 (.017) |
| MTAS Corpus, Offsets | | | | | | | | | |
| German[a] | 3 | (39) | 28 | (27) | 610 | 613 | 372,728 | .998 | .998 (.001) |
| Dutch[b] | -2 | (42) | 27 | (32) | 682 | 686 | 466,843 | .998 | .998 (.001) |
| English[c] | 28 | (317) | 66 | (311) | 606 | 692 | 372,907 | .889 | .880 (.018) |
| Semi-Spontaneous Speech Corpora[f], Onsets | | | | | | | | | |
| German[d] | -3 | (61) | 30 | (53) | 24,234 | 24,230 | 586,603,308 | .999 | .999 (.001) |
| English[e] | 5 | (118) | 34 | (113) | 48,197 | 48,205 | 2,321,038,189 | .999 | .999 (.001) |
| Semi-Spontaneous Speech Corpora[f], Offsets | | | | | | | | | |
| German[d] | 5 | (71) | 38 | (60) | 24,226 | 24,225 | 586,272,117 | .999 | .999 (.001) |
| English[e] | 11 | (131) | 50 | (121) | 48,182 | 48,199 | 2,320,005,616 | .999 | .999 (.001) |

[a] *N* = 1155 (13.8% of all data included in evaluation)

[b] *N* = 746 (10.5 %)

[c] *N* = 682 (8.7 %)

[d] *N* = 458 (5.2 %)

[e] *N* = 456 (4.4 %)

[f] The *SD*s per annotator and their covariance are substantially higher in the semi-spontaneous speech corpora, as the recordings are longer, yielding a larger range of annotated time stamps

**Table 3** Mean utterance onset times (ms) in the simple and complex conditions of the word-naming task reported in Rastle and Davis (2002) as established by a human rater (R&D), by AlignTool, by SayWhen, and by Chronset

| | Simple | | Complex | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | *M* | *SE* | *M* | *SE* | $M_{\text{diff}}$ | *t1*(20)[a] | | *t2*(38) | |
| R&D (incl. all ppts.) | 371 | 13 | 362 | 11 | 9 | 2.89 | ** | 2.02 | (*) |
| R&D | 370 | 13 | 362 | 12 | 8 | 2.67 | * | 1.93 | (*) |
| AlignTool | 334 | 11 | 332 | 11 | 2 | 0.51 | n.s. | 0.28 | n.s. |
| AlignTool - R&D | -36 | 7 | -30 | 6 | -6 | -1.68 | n.s. | -1.10 | n.s. |
| SayWhen | 439 | 16 | 448 | 16 | -9 | -1.67 | n.s. | -1.92 | (*) |
| SayWhen - R&D | 69 | 8 | 86 | 10 | -17 | -3.61 | ** | -4.67 | *** |
| Chronset | 378 | 14 | 506 | 17 | -128 | -17.04 | *** | -24.45 | *** |
| Chronset - R&D | 8 | 4 | 144 | 8 | -135 | 17.09 | *** | -32.76 | *** |

(*) $p < .1$, * $p < .05$, ** $p < .01$, *** $p < .001$

[a] *df* = 21 for Rastle (incl. all ppts)

in response times was not affected significantly by condition (see Table 3).

SayWhen allocated the onset times about 77 ms later than Rastle and Davis had done, yielding a difference between conditions in the opposite direction of what Rastle and Davis had found (-9 ms). This effect approached significance in the by-items analysis ($p$ = .063) but was not significant in the by-participants analysis. The discrepancy between the manual annotation and that established using SayWhen was particularly pronounced in the complex condition, yielding a significant effect of condition on the difference between the two types of annotation (see Table 3).

Chronset suffered from a parallel problem: While it was better than SayWhen in detecting the onsets in the simple condition, it was as inaccurate as SayWhen in the complex condition, yielding a substantial effect of condition in the opposite direction of that seen with the manual annotations and a highly significant effect of condition on Chronset's deviation from the manual annotations. Chronset's and SayWhen's results correspond to those obtained with a threshold based voice-key by Rastle and Davis (2002).

The results obtained with Chronset highlight the relevance of a post-hoc correction: Unlike a threshold-based voice-key or a fully automatic software voice-key like Chronset, SayWhen and AlignTool allow the user to correct manually the automatic annotations, eventually yielding much higher levels of accuracy. All in all, our findings suggest that the preliminary analyses of the response times provided by AlignTool are more accurate overall than those obtained by SayWhen, Chronset or a voice-key.

**MTAS corpus** For the MTAS corpus, we carried out two sets of analyses. First, we compared manual annotations of the utterance onset times with those obtained by the voice-key employed during data collection, by SayWhen, by Chronset, and by AlignTool, respectively. In the second analysis, we compared the annotations generated by AlignTool with the manual annotations for onset and offset times of words within utterances, taking into account the similarity of the last phoneme of the first and the first phoneme of the second word in pairs of successive words.

Table 4 shows how many data points had to be excluded in each language because of recording problems, participant errors, voice-key malfunction, and AlignTool malfunction. In the Dutch data set, the recordings of the word-reading task were faulty in two participants, requiring us to exclude the corresponding data points from further analysis. Also, the voice-key was very sensitive in the Dutch and English experimental setup, causing it to be triggered too early on a substantial number of trials that were excluded from the analysis. In the English data set, many additional trials were lost by a malfunction of AlignTool. As the audio quality of the English data was rather poor, AlignTool failed to annotate automatically about 9 % of the data. One would, of course, be able to annotate these trials manually. For the purpose of the present evaluation, however, we simply excluded them.

Unsurprisingly, the analysis of utterance onset times yielded largely parallel patterns of results for the picture-naming and the word-reading task (see Table 5). In German, the measurements generated by AlignTool differed least from the manual measurements, compared to SayWhen, Chronset, and the voice-key. For Chronset, there was only a small difference from the manual annotations in the object-naming task, but that difference was almost twice as big for the word-reading task. Note that there were only half as many trials in the word-reading than in the object-naming task, so a few larger deviations would impact more on the mean in the word-reading task than in the object-naming task. The voice-key tended to be triggered about 75 ms too late. SayWhen allocated the onset times about 160 ms too early, yielding the greatest deviation from the manual measurements.

**Table 4** Number of word onset and offset times lost in each language due to participant or measurement errors

| | German | | Dutch | | English | |
|---|---|---|---|---|---|---|
| No. of words tested | 9,000 | | 9,000 | | 9,000 | |
| Data points lost due to recording problems and manual annotation problems | 17 | 0.19 % | 652 | 7.24 % | 20 | 0.22 % |
| Data points lost due to participant errors | 109 | 1.21 % | 158 | 1.76 % | 161 | 1.79 % |
| Data points lost due to voice-key malfunction | 507 | 5.63 % | 1,109 | 12.32 % | 943 | 10.48 % |
| Data points lost due to AlignTool malfunction | 282 | 3.13 % | 308 | 3.42 % | 796 | 8.84 % |
| Data points remaining for analysis | 8,085 | 89.83 % | 6,773 | 80.63 % | 7,080 | 85.30 % |

In Dutch and English, the voice-key tended to generate the most exact measurements. In Dutch, AlignTool established the utterance onset times about 40 ms too early, i.e., it was too sensitive. By contrast, SayWhen and Chronset were not sensitive enough, establishing the utterance onset times 20–30 ms too late. Overall, SayWhen and Chronset deviated

**Table 5** Mean utterance onset times (in ms) in the picture naming and word-reading task in German, Dutch, and English, as established by human raters (hand annotation), by AlignTool, by a voice-key, by SayWhen, and by Chronset

| | Picture naming | | | | Word reading | | | |
|---|---|---|---|---|---|---|---|---|
| | *M* | *SE* | | | *M* | *SE* | | |
| German | | | | | | | | |
| Hand annotation | 727 | 3 | | | 724 | 4 | | |
| AlignTool | 719 | 3 | | | 704 | 4 | | |
| VK | 810 | 3 | | | 796 | 4 | | |
| SayWhen | 569 | 3 | | | 561 | 3 | | |
| Chronset | 753 | 4 | | | 771 | 4 | | |
| AlignTool minus Hand | | | -8 | *** | | | -20 | * |
| VK minus Hand | | | 83 | *** | | | 72 | *** |
| SayWhen minus Hand | | | -158 | *** | | | -163 | *** |
| Chronset minus Hand | | | 26 | *** | | | 47 | *** |
| Dutch | | | | | | | | |
| Hand annotation | 739 | 4 | | | 664 | 5 | | |
| AlignTool | 707 | 4 | | | 615 | 6 | | |
| VK | 740 | 4 | | | 659 | 5 | | |
| SayWhen | 770 | 4 | | | 697 | 5 | | |
| Chronset | 749 | 4 | | | 698 | 6 | | |
| AlignTool minus Hand | | | -32 | *** | | | -50 | *** |
| VK minus Hand | | | 1 | n.s | | | -5 | $^{(*)}$ |
| SayWhen minus Hand | | | 31 | *** | | | 33 | *** |
| Chronset minus Hand | | | 11 | *** | | | 34 | *** |
| English | | | | | | | | |
| Hand annotation | 735 | 4 | | | 718 | 4 | | |
| AlignTool | 789 | 4 | | | 772 | 4 | | |
| VK | 753 | 4 | | | 729 | 4 | | |
| SayWhen | 1073 | 19 | | | 819 | 5 | | |
| Chronset | 562 | 7 | | | 673 | 10 | | |
| AlignTool minus Hand | | | 53 | *** | | | 54 | *** |
| VK minus Hand | | | 18 | *** | | | 11 | *** |
| SayWhen minus Hand | | | 338 | *** | | | 101 | *** |
| Chronset minus Hand | | | -181 | *** | | | -51 | *** |

$^{(*)}$ $p < .1$, * $p < .05$, ** $p < .01$, *** $p < .001$

substantially less from the manual measurements in Dutch than in German, possibly due to the fact that the recording quality was better for the Dutch than for the German speakers. In line with this interpretation of the Dutch data, SayWhen's and Chronset's performance dropped markedly with the English recordings, which were of by far the poorest quality overall. AlignTool was able to deal with this problem reasonably well, yielding much smaller deviations from the manual annotations than SayWhen and Chronset. The voice-key was as exact for the English speakers as for the Dutch speakers. Recall that the audio signal was first fed to the external voice-key (Hasomed NesuBox 2) and was then recorded. The relatively unaffected operation of the voice-key along with rather poor recordings suggest that the recording problems in the English corpus arose after the voice-key had operated.

Overall, these findings suggest that AlignTool establishes utterance onset times reasonably exactly, irrespective of the quality of the recordings in terms of the signal-to-noise ratio. However, users will need to tune the parameters to their recording quality. In Appendix A6 of the User Manual (Belke et al., 2017), we give some advice on how to do this. The present results also indicate that users need to edit some of the automatically generated measurements manually in order to obtain optimal results. To this end, they can access the TextGrid files established by AlignTool and edit them directly. All changes made can be saved and imported to the Excel workbook by means of the *Import to TextGrids* function.

Table 6 presents the average deviation from the manual annotations at the transitions of successive words in the four-word utterances generated in the reading task. These transitions pertained to the last phoneme of the first word and the first phoneme of the second word and accordingly to the offset and onset phonemes of the second and third word and the third and fourth word. Table 7 presents the results of the statistical

**Table 6** Measurement differences between AlignTool and manual annotators for dissimilar, similar, and identical transitions between the first and second, second and third, and third and fourth word in the word-reading task

| | Dissimilar | | | | | Similar | | | | | Identical | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Position | $M^a$ | $SE^a$ | $M^b$ | $SE^b$ | $N$ | $M^a$ | $SE^a$ | $M^b$ | $SE^b$ | $N$ | $M^a$ | $SE^a$ | $M^b$ | $SE^b$ | $N$ |
| | Onsets | | | | | | | | | | | | | | |
| | German | | | | | | | | | | | | | | |
| 1-2 | -1 | 2 | | | 632 | -4 | 28 | | | 444 | -12 | 7 | | | 278 |
| 2-3 | 25 | 20 | | | 618 | 16 | 13 | | | 433 | -8 | 3 | | | 302 |
| 3-4 | 77 | 24 | | | 652 | 81 | 26 | | | 480 | 21 | 17 | | | 220 |
| | Dutch | | | | | | | | | | | | | | |
| 1-2 | 0 | 6 | -6 | 5 | 496 | -1 | 7 | -10 | 5 | 304 | 4 | 9 | -6 | 4 | 308 |
| 2-3 | 44 | 13 | 12 | 4 | 394 | 64 | 21 | 4 | 4 | 361 | 7 | 15 | -25 | 4 | 352 |
| 3-4 | 537 | 55 | 18 | 5 | 466 | 349 | 50 | 2 | 4 | 403 | 409 | 78 | -28 | 6 | 239 |
| | English | | | | | | | | | | | | | | |
| 1-2 | 26 | 3 | | | 337 | 10 | 4 | | | 255 | 35 | 12 | | | 548 |
| 2-3 | 35 | 19 | | | 431 | 33 | 21 | | | 293 | 31 | 12 | | | 416 |
| 3-4 | 99 | 27 | | | 424 | 30 | 25 | | | 322 | 87 | 25 | | | 392 |
| | Offsets | | | | | | | | | | | | | | |
| | German | | | | | | | | | | | | | | |
| 1-2 | 6 | 3 | | | 632 | -4 | 29 | | | 444 | -11 | 8 | | | 278 |
| 2-3 | 36 | 20 | | | 618 | 15 | 11 | | | 433 | -5 | 3 | | | 303 |
| 3-4 | 59 | 20 | | | 653 | 40 | 15 | | | 480 | 9 | 6 | | | 220 |
| | Dutch | | | | | | | | | | | | | | |
| 1-2 | -1 | 7 | -7 | 6 | 496 | -12 | 8 | -20 | 7 | 304 | -2 | 8 | -11 | 5 | 308 |
| 2-3 | 30 | 9 | 8 | 4 | 394 | 32 | 9 | 4 | 4 | 362 | -10 | 7 | -30 | 4 | 352 |
| 3-4 | 143 | 19 | 16 | 4 | 465 | 72 | 15 | -6 | 4 | 403 | 87 | 26 | -20 | 5 | 239 |
| | English | | | | | | | | | | | | | | |
| 1-2 | 22 | 4 | | | 337 | -5 | 5 | | | 255 | -4 | 4 | | | 548 |
| 2-3 | 29 | 18 | | | 431 | 4 | 22 | | | 293 | -16 | 12 | | | 416 |
| 3-4 | 59 | 23 | | | 425 | 20 | 22 | | | 323 | -3 | 17 | | | 392 |

[a] *M* and *SE* of the first annotation of the trials in full length

[b] *M* and *SE* of the annotation of the trimmed trials (Dutch data only)

**Table 7** Results of the statistical analyses of the effects of Transition Position, Transition Similarity, and their interaction on the differences between the measurements obtained with AlignTool and manual annotations

| | *df1* | *F* | | *F* | | *F* | | *F* | |
|---|---|---|---|---|---|---|---|---|---|
| | Word onsets | | | | | | | | |
| | | German (*df2* = 4,050) | | Dutch (*df2* = 3,314) (annotation of full trials) | | Dutch (*df2* = 3,314) (annotation of trimmed trials) | | English (*df2* = 3,409) | |
| Transition Position | 2 | 6.96 | *** | 127.64 | *** | .86 | n.s. | 5.29 | ** |
| Transition Similarity | 2 | 1.74 | n.s | 2.59 | (*) | 25.88 | *** | 1.83 | n.s. |
| Position × Similarity | 4 | 0.35 | n.s. | 3.15 | * | 6.89 | *** | 0.93 | n.s. |
| | Word offsets | | | | | | | | |
| | | German (*df2* = 4,052) | | Dutch (*df2* = 3,314) (annotation of full trials) | | Dutch (*df2* = 3,314) (annotation of trimmed trials) | | English (*df2* = 3,411) | |
| Transition Position | 2 | 3.13 | * | 51.87 | *** | 2.47 | n.s. | 1.60 | n.s. |
| Transition Similarity | 2 | 2.73 | (*) | 5.58 | ** | 20.96 | *** | 6.64 | *** |
| Position × Similarity | 4 | 0.19 | n.s. | 3.56 | ** | 6.08 | *** | 0.40 | n.s. |

analysis of the effects of transition position, transition similarity, and their interaction on the difference between the measurements established by AlignTool and the manual measurements. The position of the transition within the four-word utterances had a substantial effect on the accuracy of the measurements generated by AlignTool (or MAUS, to be precise), with measurement accuracy decreasing substantially across the four-word sequence (see Table 6). Indeed, from a technical perspective one would expect that accuracy decreases with increasing length, as the search space (i.e., all possible segmentations) of the HMM-based alignment algorithm increases quadratically with utterance length, thus increasing the probability of errors at later positions within the utterance. The position effect was particularly pronounced in the Dutch data, yielding average deviations of 400–500 ms for the transition from the third to the fourth word. Given that the recording quality of the Dutch data was excellent, this finding is surprising and clearly exceeds the technically induced position effect caused by the increase in length.

Inspection of trials yielding such high deviations between the third and the fourth word suggested that *segmentSpeech* had malfunctioned on some occasions, taking audible

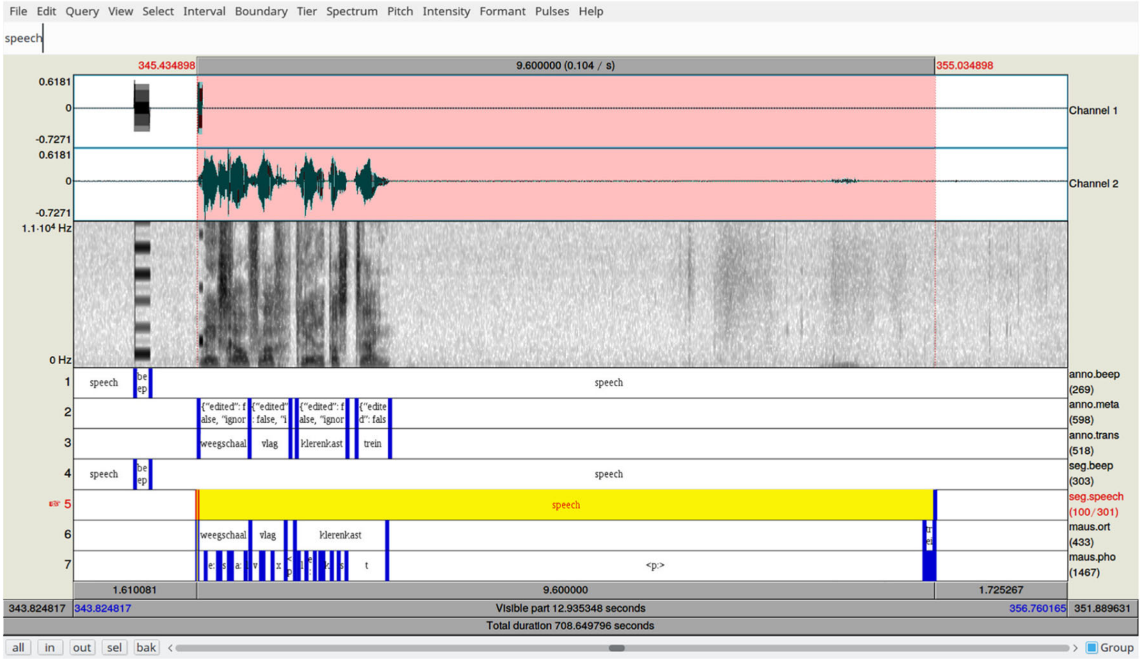

File Edit Query View Select Interval Boundary Tier Spectrum Pitch Intensity Formant Pulses Help
speech
345.434898
9.600000 (0.104 / s)
355.034898
0.6181
0
-0.7271
Channel 1
Channel 2
1.1·10⁴ Hz
0 Hz
1 speech be ep speech anno.beep (269)
2 {"edited": false, "ignor {"edited": false, "i {"edited": false, "ignor {"edited": fals anno.meta (598)
3 weegschaal vlag klerenkast trein anno.trans (518)
4 speech be ep speech seg.beep (303)
5 speech seg.speech (100/301)
6 weegschaal vlag klerenkast maus.ort (433)
7 e: s a: v x k t <p:> maus.pho (1467)
1.610081
9.600000
1.725267
343.824817
343.824817
Visible part 12.935348 seconds
356.760165
351.889631
Total duration 708.649796 seconds
all in out sel bak
Group

**Fig. 5** Example of a trial from the Dutch MTAS (Manually Temporally Annotated Speech) corpus used in the evaluation of AlignTool: in *segmentSpeech*, AlignTool has erroneously established the utterance offset time too late, impacting on the accuracy of the subsequent *alignMAUS* processing step

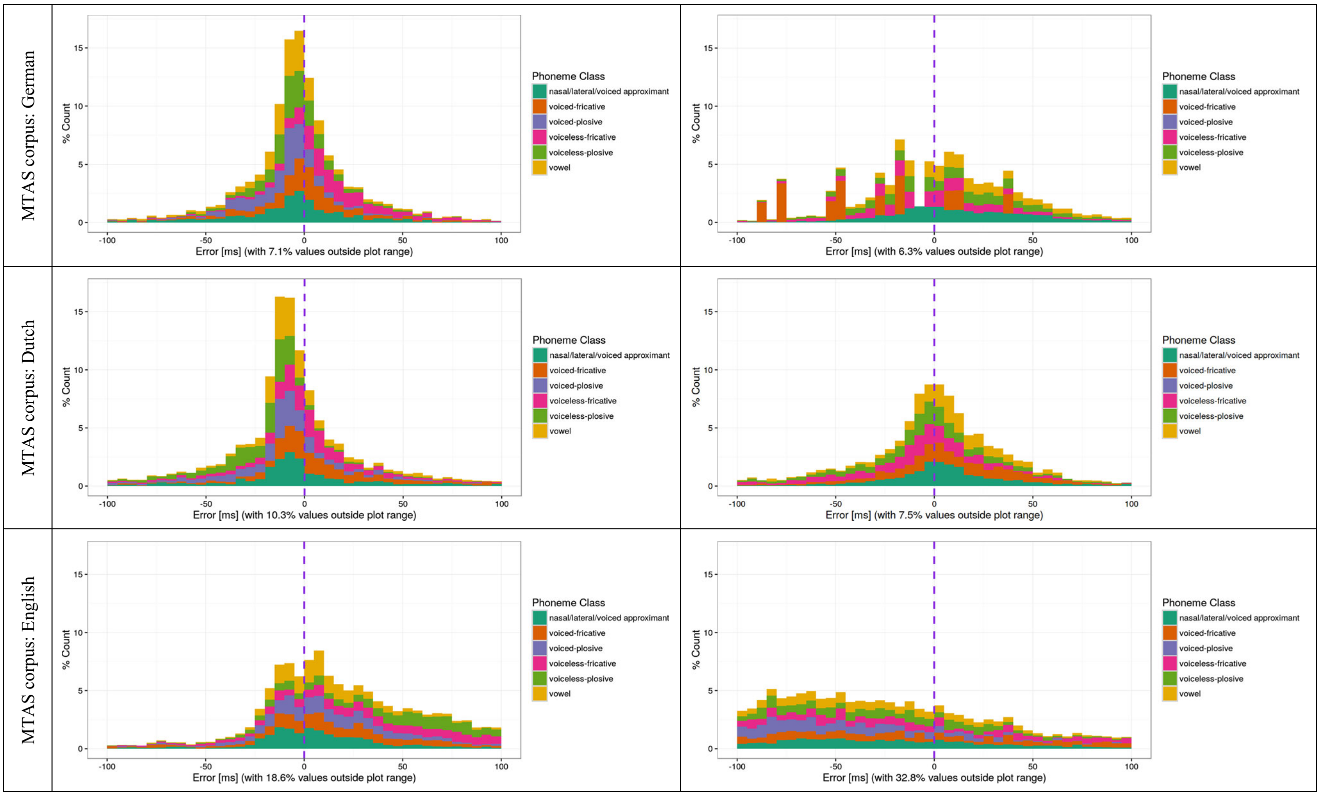


**Fig. 6** Distribution of measurement differences between -100 ms and +100 ms by phoneme types in onsets (left) and offsets (right) in the German, Dutch, and English sections of the MTAS (Manually Temporally Annotated Speech) corpus

breathing of the participants towards the end of the trial to be speech rather than noise (see Fig. 5 for an example trial). This is caused by the automatic floor noise detection used to establish the silence threshold. The excellent audio quality of the Dutch recordings led to a low floor noise and thus breathing was likely more prominent than in recordings with a poor signal-to-noise ratio. This problem may have occurred on other trials as well. To assess this, we trimmed all audio data, deleting the interval starting 500 ms after utterance offset (as established in the manual annotation) and ending at the end of the trial.[3] Table 6 presents the average deviations in onsets and offsets by position and transition similarity. Trimming the audio data improved the results of the automatic annotation considerably, eliminating the statistical effect of transition position (see Table 7). There was still an effect of transition similarity, with the offset and onset times of dissimilar and similar words being annotated a little earlier than the manual annotations early on in the utterances and slightly later than the manual annotations at later positions in the utterance. This effect was reversed, however, for transitions with identical phonemes at the end of the first and the beginning of the second word, yielding a significant interaction of transition similarity and transition position. It is not clear why the position effect reversed for this transition similarity only but it is important to keep in mind that with identical transitions, manual annotations are largely arbitrary (see Appendix 3 for the annotation guidelines we followed). For the time being, the most important finding is that trimming the data improved the performance of AlignTool considerably, suggesting that *segmentSpeech* had not segmented the speech-relevant intervals in the trials reliably. This reflects a trade-off between setting highly sensitive parameters and sacrificing onset and offset accuracy.

Figure 6 presents the deviations from the manual annotations for the onset and offset times by phoneme type. For the Dutch MTAS corpus, Fig. 6 shows the results of the analysis of the trimmed audio data. There was a clear connection between the quality of the audio recordings and the overall amount of deviation of the annotation generated by AlignTool from the manual annotation. While there was only little deviation from the manual annotations in Dutch (excellent recording quality), there was more deviation in German (average recording quality) and most in English, where the recordings were worst. The deviation seen across languages was not systematically

[3] We are, of course, aware that users of AlignTool would normally not have the manually annotated utterance offset times at their disposal to implement the solution we used. However, upon inspecting the data generated by AlignTool, they would typically spot outliers in the distributions of onset times, offset times and word durations (by position) and would thus be able to identify trials with potential problems of the kind observed here. Users would then be able to trim the audio files manually or automatically, using Praat. Note that such trimming does not need to be exact, as long as it cuts off large sections of the non-speech phase at the end of the trial.

affected by phoneme type – positive and negative deviations from the manual annotations were seen in all phoneme classes alike.

**Semi-spontaneous speech** In the English and German corpora of semi-spontaneous speech, we had to exclude some of the data from the analysis. As these corpora involved dialogue-like settings, speakers occasionally spoke at the same time. These sections of the corpus were excluded, as MAUS is unlikely to be able to deal with them. Interjections, like "hm", speech errors, and incomprehensible words were excluded from the analysis as well. Finally, all trials associated with AlignTool malfunction were excluded, leaving 75.4 % of the words in the Map Task corpus and 82 % of the words in the Comparison Task corpus for analyses. There was no need to exclude any data from the Sjerps&Meyer corpus.

Table 8 presents the average measurement deviation in the corpora and the correlation between this measurement deviation and the position of a word in the participants' utterance. Unlike what we had seen in the analyses of the MTAS data, there was no consistent positive correlation between the position of a word in the utterance and AlignTool's accuracy; the correlation was positive for the Dutch data, negative for the German data and absent for the English data. In order to interpret these correlations, we also assessed the absolute difference between the annotations by AlignTool and the manual raters (see Table 8). For the Display Comparison Task corpus, AlignTool seems to have annotated the relevant word boundaries too early, and this error intensified over positions. As a result, there was a significant negative correlation between utterance position and the average differences between AlignTool and the manual annotations, along with a positive correlation between utterance position and the absolute differences between AlignTool and the manual annotations. The Dutch data suggest that AlignTool again tended to annotate relevant word boundaries earlier than the manual annotators. Over the four positions in the Dutch utterances, this difference grew more positive, accounting for the positive correlation between the average differences and utterance positions. For the absolute differences, there was no significant correlation with utterance position, as the absolute differences would first be positive, then move towards 0 and finally above 0 again, yielding a non-linear sequence of decreasing and increasing absolute differences. Finally, in the English Map Task corpus, there was no systematic effect of utterance position on the average differences between the annotations generated by AlignTool and by the manual annotators. In fact, the analysis of the absolute differences showed that the error became smaller over utterance positions rather than bigger.

It is noteworthy that the utterance length in the three corpora of semi-spontaneous speech differed considerably. For Dutch, participants produced utterances of the type "Put the A above (below) the B and put the C below (above) the D", including 13 words in total. For the Map Task corpus, by contrast, there were, on average, 2 min and 40 s of pure speech in each of the 21 recordings, which we analyzed. For the Display Comparison Task corpus, the recordings were even longer: on average, there were 7 min and 30 s of pure speech from each speaker pair. Hence, one might expect that the effect of word position on average absolute differences seen in the English and German data decreases considerably when the recordings are pre-segmented into shorter sections of, say, 30 s each, prior to applying *alignMAUS* to the recordings (see Belke et al., 2017).

Figure 7 presents the deviations from the manual annotations for the onset and offset times by phoneme type. In German and English, the results mirrored those obtained for the MTAS corpus: While the quality of the audio recordings, which was excellent in the English Map Task corpus but much poorer in the German Display Comparison Task corpus, clearly impacted on the overall deviation of the annotation generated by AlignTool from the manual annotation, there was no systematic effect of phoneme type. By contrast, the results for the Dutch corpus differed from those obtained for the MTAS corpus in that there was a marked effect of phoneme type, especially for word onsets. AlignTool annotated the onsets of words starting with plosives substantially earlier than the human raters, whereas all other phoneme types were annotated similarly by AlignTool and the human raters. For offsets

**Table 8** Semi-spontaneous speech: Mean measurement differences (in ms) for word onsets and offsets and correlations of the magnitude of the deviation with utterance position. For each corpus, the first row presents the average differences, including those greater and smaller than 0, and the second row presents the average absolute differences

| | | | Word onsets | | | | Word offsets | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | *N* | *M* | *SE* | $r_{\text{deviation} \times \text{position}}$ | | *M* | *SE* | $r_{\text{deviation} \times \text{position}}$ | |
| Display Comparison Task corpus (German) (Sichelschmidt et al., 2010) | Average difference | 8,879 | -129 | 13 | -.074 | *** | 23 | 13 | -.065 | *** |
| | Absolute difference | | 393 | 13 | .097 | *** | 404 | 13 | .090 | *** |
| Sjerps&Meyer corpus (Dutch, nouns only) (Sjerps & Meyer, 2015) | Average difference | 3,971 | -23 | 6 | .224 | *** | -14 | 6 | .164 | *** |
| | Absolute difference | | 108 | 6 | -.022 | n.s. | 96 | 6 | -.011 | n.s. |
| Map Task corpus (English) (Anderson et al., 1991) | Average difference | 10,272 | -7 | 11 | .009 | n.s. | -6 | 11 | .009 | n.s. |
| | Absolute difference | | 239 | 11 | -.030 | ** | 246 | 11 | -.029 | ** |

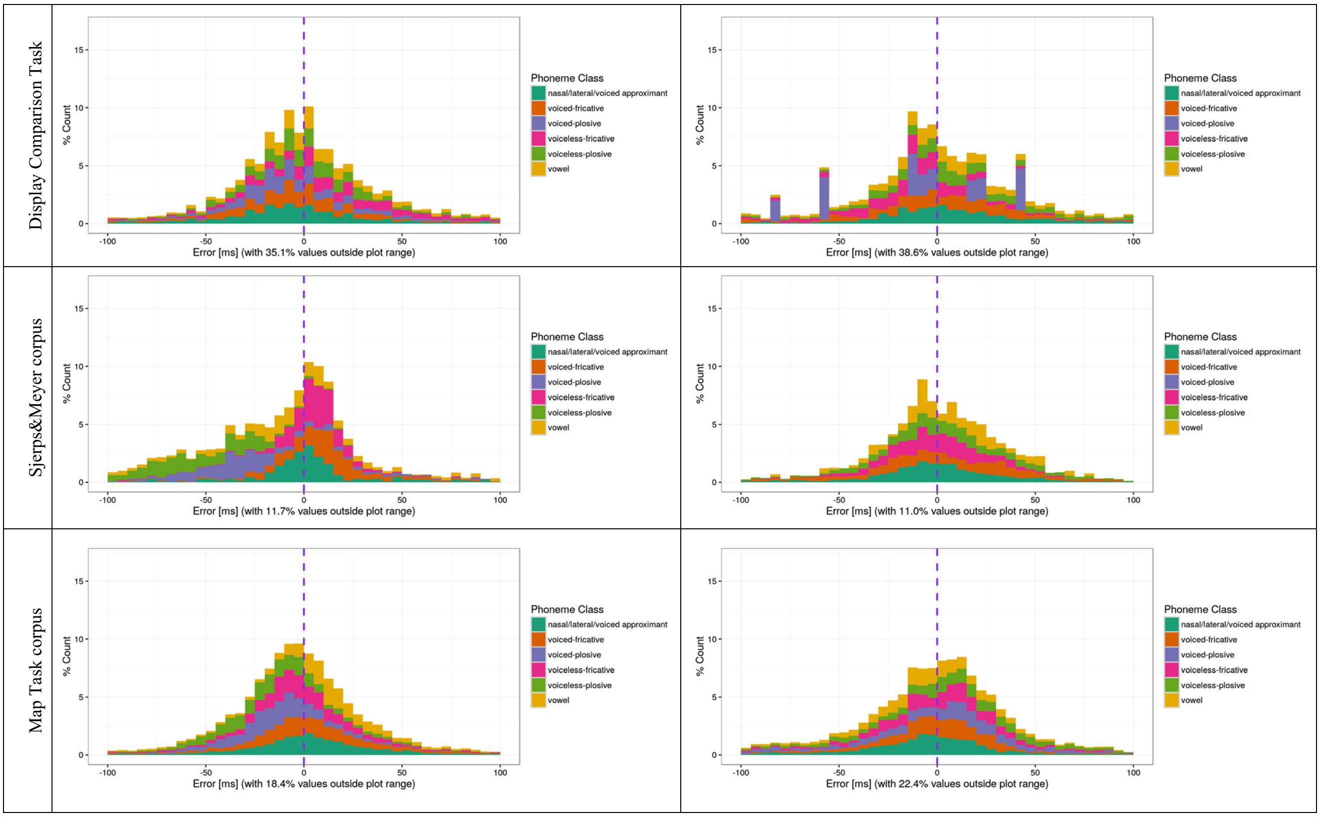


**Fig. 7** Distribution of measurement differences between -100 ms and +100 ms by phoneme types in onsets (left) and offsets (right) in the corpora of semi-spontaneous speech in German (Display Comparison Task corpus), Dutch (Sjerps&Meyer corpus), and English (Map Task corpus)

this effect was not visible. We assume that the systematic deviations in the onsets came about because the human raters had used the moment of plosion in order to establish the speech onset whereas AlignTool established the beginning of the plosives slightly earlier. This difference impacts on the temporal annotation of onsets only, as for plosives in the word offsets, both AlignTool and the human raters established the moment of plosion as the offset of the word.

## Outlook

AlignTool is an open source tool for the semi-automatic temporal alignment of speech in single- and multiple-word utterances and semi-spontaneous speech. In a large-scale evaluation, we have identified strengths and weaknesses of AlignTool, demonstrating that it can provide most accurate automatic alignments for recordings with an excellent signal-to-noise ratio but becomes less accurate as the recording quality decreases.

Evidently, each researcher will try to ensure that the recordings are of the best possible quality in terms of their signal-to-noise ratio, but in our experience, ideal recording conditions are rarely given. Therefore, we configured AlignTool to perform with recordings of poorer quality as well and to allow for easy-to-implement manual corrections in Praat. However, AlignTool is likely to perform less robustly and less accurately with audio signals of poorer quality, requiring the user to correct more trials than with audio signals of better quality.

In language production research, AlignTool can be used as a digital voice-key as well as as a tool for establishing word onset and offset times in more complex, semi-spontaneous settings. Our pilot data from evaluating AlignTool's accuracy in automatically aligning semi-spontaneous speech are promising in that the deviations were small, especially for recordings of excellent quality, and there was no systematic effect of a word's position in the utterance on alignment accuracy. Note that MAUS deserves most of the credit for this, as the alignments of semi-spontaneous speech largely relied on MAUS.

Unlike for the semi-spontaneous speech, we found that the automatic analyses of the four-word utterances in the MTAS corpus with AlignTool exhibited a substantial effect of word position. We presume that this contrast between the two types of utterances came about because the recordings in the MTAS corpus included long silent intervals, namely the interval between stimulus onset and response onset, when participants were planning their utterance, and the interval after utterance completion until the beginning of the next trial. We have demonstrated for the Dutch data that even in cases when the recording quality is excellent, non-speech sounds can have an impact on the quality of the automatic alignments generated by AlignTool. One might reduce such problems by training the automatic speech recognition system to distinguish between speech and non-speech sounds. Note, however, that such training is necessarily tied to the given recording scenario and is therefore unlikely to transfer to other recording scenarios.

Given that users can correct the results of AlignTool manually, the alignments provided by AlignTool can be potentially as accurate as those generated by hand. However, the aim is, of course, to have AlignTool generate as many word onsets on its own as possible. To this end, we recommend that users prepare manual annotations of a sample of the utterances they want to align using AlignTool and use them to find the optimal parameter settings for their recording setting. In Appendix A6 of the User Manual, we give some recommendations on how to do this. In all likelihood, an optimal parameter setting for a given recording scenario can be carried forth to new recordings. Therefore, it is worthwhile to invest some tuning effort when first using AlignTool.

However, our evaluation results also indicate that even when parameters are optimized, users must not rely on the results generated by AlignTool blindly but need to be intelligent inspectors of its results. For instance, by generating histograms of the onset times and the durations of individual words, users can identify apparent outliers so as to find out whether there are problems in the recordings of the kind reported for the Dutch section of the MTAS corpus, where a breathing noise was mistaken for speech and distorted the automatic alignment.

Apart from the domains we have evaluated AlignTool for in this paper, it can also be used in research on language comprehension, where it may be of interest to link the temporal structure of spoken utterances to listeners' eye movements, for instance in visual word experiments or in instruction settings like the Map Task setting. In addition, AlignTool can be applied to analyzing the onset and offset times of pseudowords, as long as the pseudowords are phonotactically plausible in the speakers' language, such that the G2P-service (BAS, 2017b) can generate a pronunciation based on the pseudowords' orthography. We have tested AlignTool on short sequences of pseudowords used in an artificial language learning study with German speakers (Bebout & Belke, 2017). Each of the utterances consisted of four or eight pseudowords (cf. the prose and rhyme training conditions in the study). There were no manual annotations of the onset and offset times of the pseudowords but we assessed the outcome of AlignTool's measurements visually for a sample of the 144 recordings we aligned in this way and found the results to be accurate.

Moving on from AlignTool, the next big challenge will be to develop efficient tools for the (semi-)automatic annotation of speech recorded in dialogue settings involving multiple speakers. Rosenfelder et al. (2011) have presented FAVE-align, a tool that allows users to temporally align speech recorded from multiple speakers in dialogue settings, such as sociolinguistic interviews. Users transcribe each speaker's utterance in a separate tier and feed this information to FAVE-align, which performs forced alignments using the Penn Phonetics Lab Forced Aligner (P2FA). The tool has not been evaluated for temporal accuracy, but given that it allows for manual corrections of the alignment very high levels of accuracy should be achievable.

In sum, many psycholinguistic studies require precise information about the time course of spoken utterances. AlignTool is an open source instrument that should, we hope, support researchers in the semi-automatic analysis of their corpora. By functioning as a voice-key as well as as a tool for the analysis of word onset and offset times in more complex utterances, we expect that AlignTool will open new avenues in language production research.

**Author note** This research was funded by grant no. BE 3176/4-1, awarded to Eva Belke, Britta Wrede, and Antje Meyer by the German Research Foundation (DFG). We thank Andrea Krott for making her laboratory facilities available to us in order to collect the data for the British English section of the MTAS corpus. We are most grateful to Theodor Berwe, Sabrina Böckmann, Saskia Bohemann, Maria Hosfeld, Anastasia Los, Sara Kattanek, Mareike Klamandt, Natalie Kroll, and Esther Seyffarth for their help in measuring manually the onset and offset times of the words and to Jeroen van Paridon for carrying out the analyses with SayWhen. AlignTool and its documentation are available at https://www.linguistics.ruhr-uni-bochum.de/~belke/aligntool.shtml

## Appendix 1

Automatic speech recognition systems work by training the parameters of a statistical model – in this case a Hidden-Markov-Model (HMM) – with data from an annotated set of speech data (Rabiner, 1989). HMMs model sub-word-units, normally phone-like segments, through a sequence of unobservable states associated with observable features modelled by mixtures of Gaussian distributions. These sub-word-units can then be concatenated to form words. The acoustic features in such standard approaches are based on the spectral information of the speech signal where only the information of the vocal tract configuration is retained while discarding information about the sound source (i.e. the fundamental frequency; Young, 1996).

In order for such a recognition system to perform well on new data, several parameters need to be optimized on the basis of a cross-evaluation test set that is independent of the training set. Optimizing these parameters requires weighting, for example, bottom-up acoustic information against top-down information as provided by a statistical language model or a grammar, or determining the number of Gaussian distributions needed to optimally model the training data while maintaining sufficient generalisation power. Other parameters relate to optimizing the ability of the system to predict the (prior) probabilities of words. In addition, a speech recognition system designed to recognize speech on its own needs a custom-made lexicon – a predefined set of words to be recognized. Words unknown to the system are very difficult to spot as they will be projected onto the next most similar word. In addition, the system needs to be trained with sequences of words that are likely to occur in the application scenario. This means that prior to the use in a specific scenario, it is necessary to train a statistical language model with data from this scenario.

However, the system's search space can be reduced dramatically by providing it with an orthographic transcription of the utterances. Given such a transcription, the automatic speech recognition system will force-align the given sequence of phone models (as derived from the orthographic transcription through a pronunciation lexicon or a phoneme string generated by a Grapheme-Phoneme Converter, e.g., BAS, 2017b) to the speech signal and provide the most likely temporal alignment.

## Appendix 2

### Lists of object names used in the picture naming and the word-reading task in each language.

*English:* airplane, anchor, apple, arm, ball, bed, bowl, bra, broom, cactus, chair, cheese, clock, clown, crab, cross, desk, dog, dragon, dress, drum, ear, egg, elephant, envelope, eye, fish, flag, flower, frog, genie, ghost, glasses, glove, grapes, guitar, hammer, horse, igloo, iron, jacket, key, kite, knife, ladder, leaf, map, monkey, nail, nose, octopus, onion, orange, owl, penguin, plate, plug, priest, puzzle, queen, ring, robot, ruler, saddle, shark, shoe, skateboard, skeleton, sled, slide, smoke, snake, snowman, spider, spoon, squirrel, statue, stool, strawberry, stroller, sun, swan, swing, table, tent, thermos, thumb, train, tree, umbrella, unicorn, vase, volcano, watch, whale, worm, wrench, yoyo, zebra, zipper

*Dutch:* aap, aardbei, ananas, appel, arm, auto, bed, blad, bloem, bril, brug, bus, cactus, citroen, clown, dak, deur, dolfijn, draak, eekhoorn, eend, eenhoorn, ei, emmer, eskimo, fiets, fles, fontein, geweer, gieter, glas, gordijn, graf, grasmaaier, helikopter, hoed, hond, iglo, jojo, kaas, kerk, kip, klerenkast, krokodil, kruis, ladder, leeuw, lepel, masker, muur, neus, noot, olifant, oog, oor, paard, paraplu, pijl, pleister, pruik, radio, ring, robot, saxofoon, schaar, schildpad, schommel, schroevendraaier, sigaret, sjaal, skatebord, skelet, slang, sleutel, sneeuwman, spaghetti, spiegel, ster, stofzuiger, strijkijzer, tafel, telefoon, trap, trein, trompet, uil, varken, vlag, vleermuis, vlieger, voet, vulkaan, weegschaal, wereldbol, wiel, wolk, zadel, zon, zwaan, zwembad

*German:* Anker, Apfel, Arm, Auge, Auto, Ball, Besen, Bett, Blitz, Blume, Briefkasten, Brille, Clown, Dach, Daumen, Drache, Eichhörnchen, Eimer, Einhorn, Elefant, Ente, Esel, Eule, Fass, Fenster, Fledermaus, Flugzeug, Frau, Frosch, Geweih, Giraffe, Glas, Globus, Grab, Gürtel, Handschuh, Herz, Hund, Iglu, Jacke, Junge, Käfig, Käse, Kinderwagen, Kleid, Knochen, Knopf, Kreuz, Krokodil, Leiter, Löwe, Luftballon, Mädchen, Messer, Mund, Nagel, Nase, Ohr, Ohrring, Papagei, Pfeil, Pflaster, Pistole, Rakete, Roboter, Säule, Schere, Schlange, Schlüssel, Schmetterling, Schnecke, Schneemann, Schraube, Schreibmaschine, Schuh, Schwan, Schwein, Skateboard, Skelett, Soldat, Sonne, Spargel, Spiegel, Spritze, Staubsauger, Straße, Streichholz, Stuhl, Tasse, Telefon, Tisch, Trichter, Trommel, U-Boot, Uhr, Waage, Wasserhahn, Zigarette, Zitrone, Zwiebel

## Appendix 3

### Annotation Rules

Ten different research assistants were involved in the annotation of the data. As the visible waveform information was not always reliable, we annotated the files by ear. To guarantee that the annotations were comparable, we established basic rules for coding. In single word utterances, as recorded in the object-naming task, the onset of a word was marked as soon as the first audible speech sound started. Accordingly, the offset was marked at the end of audible speech. For utterances that did not appear in isolation but in the presence of other words, additional rules applied: When two words blended into each other, we marked the first point during the utterance, at which the first word was not audible any longer as the offset of the first word. For instance, when uttering "that is", the /t/ of "that" will still be audible when the /i/ is already being uttered and vice versa. We marked the beginning of the word "is" at that point, at which only the /i/ was audible and the /t/ not anymore.

Sometimes words were recognizable in their context but hardly audible in isolation, such as "is" in "it is blue". In these cases, we chose to mark the ending of the first and the beginning of the last word and marked the space between them as the shorter word. In the above example, this means we annotated "it" and "blue" and marked the rest as "is".

We agreed that the ending of one word and the beginning of another should not overlap. Therefore, in an utterance like "it is", if "it" ended at, for instance, 12 s and 220 ms, "is" would have been marked as starting at 12 s and 221 ms. This kind of coding is useful for potential applications of the data for ASR systems, which often do not accept overlaps across words (e.g., Fink, 1999). When transitions between words were identical, as in “unicorn-nail” and participants did not pause between words, we divided the total length of the sound at the transition (/n/ in the example) in half and assigned each half to one of the words.

In the Map Task corpus (Anderson et al., 1991) and the Display Comparison Task corpus (Sichelschmidt et al., 2010), interjections, cases in which two speakers spoke simultaneously, and words that were unintelligible or distorted in other ways were highlighted so as to be excluded from the analyses with AlignTool.

## References

Abrams, L., & Jennings, D. T. (2004). VoiceRelay: Voice key operation using Visual Basic. *Behavior Research Methods, Instruments, and Computers, 36,* 771-777.

Anderson, A. H., Bader, M., Bard, E. G., Boyle, E., Doherty, G., Garrod, S., ... Weinert, R. (1991). The HCRC Map Task Corpus. *Language and Speech, 34,* 351-366.

Baayen, R. H., Piepenbrook, R., & van Rijn, H. (1995). *The CELEX lexical database* (CD-ROM). Linguistic Data Consortium, University of Pennsylvania, Philadelphia, PA.

BAS (Bavarian Archive for Speech Signals) (2017a, August 9). BAS WebServices. Retrieved from https://clarin.phonetik.uni-muenchen.de/BASWebServices/interface.

BAS (Bavarian Archive for Speech Signals) (2017b, August 9) BAS WebServices: G2P. Retrieved from https://clarin.phonetik.uni-muenchen.de/BASWebServices/interface/Grapheme2Phoneme.

BAS (Bavarian Archive for Speech Signals) (2017c, August, 8). BAS WebServices: General Help – Terms of Usage. Retrieved from https://clarin.phonetik.uni-muenchen.de/BASWebServices/help/termsOfUsage#termsofusage.

Bates, E., D’Amico, S., Jacobsen, T., Székely, A., Andonova, E., Devescovi, A., ... Tzeng, O. (2003). Timed picture naming in seven languages. *Psychonomic Bulletin and Review, 10,* 344-380.

Bebout, J. & Belke, E. (2017). Language play facilitates language learning: Optimizing the input for rapid gender-like category induction. *Cognitive Research: Principles and Implications, 2,* 11.

Belke, E., Keite, V., & Schillingmann, L. (2017). AlignTool Documentation. Retrieved from https://www.linguistics.rub.de/~belke/aligntool.shtml.

Bock, J. K. (1996). Language production: Methods and methodologies. *Psychonomic Bulletin & Review, 3,* 395-421.

Boersma, P. & Weenink, D. (2016). Praat: Doing phonetics by computer (Version 6.0.14) [Computer software]. Retrieved from http://www.praat.org/.

Brennan, S. E., Schuhmann, K. S., & Batres, K. M. (2013). Entrainment on the move and in the lab: The Walking Around Corpus. *Proceedings of the 35th Annual Conference of the Cognitive Science Society.*

Clark, H. (1996). *Using Language*. Cambridge: Cambridge University Press.

Coco, M. I., & Keller, F. (2015). Integrating mechanisms of visual guidance in naturalistic language production. *Cognitive Processing, 16,* 131-150.

Coco, M. I., Malcolm, G. L., & Keller, F. (2014). The interplay of bottom-up and top-down mechanisms in visual guidance during object naming, *The Quarterly Journal of Experimental Psychology, 67,* 1096-1120.

Duyck, W., Anseel, F., Szmalec, A., Mestdagh, P., Tavernier, A., & Hartsuiker, R. (2008). Improving accuracy in detecting acoustic onsets. *Journal of Experimental Psychology: Human Perception & Performance, 34,* 1317-1326.

Fink, G. A. (1999). Developing HMM-based recognizers with ESMERALDA. In V. Matousek, P. Mautner, J. Ocelíková, & P. Sojka (Eds.), *Lecture notes in artificial intelligence science: Vol. 1692. Text, speech and dialogue: Second international workshop, TSD ’99*, Plzen, Czech Republic, September 13-17, 1999 (pp. 229-234). Berlin: Springer.

Forster, K. I., & Forster, J. C. (2003). A windows display program with millisecond accuracy. *Behavior Research Methods, Instruments, & Computers, 35,* 116-124.

Fox Tree, J. E., & Clark, H. H. (1997). Pronouncing “the” as “thee” to signal problems in speaking. *Cognition 62*, 151-167.

Griffin, Z. M., & Bock, J. K. (2000). What the eyes say about speaking. *Psychological Science, 11,* 274-279.

Griffin, Z. M., & Ferreira, V. S. (2006). Properties of spoken language production. In M. J. Traxler, & M. A. Gernsbacher (Eds.), *Handbook of psycholinguistics* (2nd ed.) (pp. 21-59). London: Elsevier.

Hanna, J. E., & Brennan, S. E. (2007). Speakers' eye gaze disambiguates referring expressions early during face-to-face conversation. *Journal of Memory and Language, 57,* 596-615.

Hüttig, F., Rommers, J., & Meyer, A. S. (2011). Using the visual world paradigm to study language processing: A review and critical evaluation. *Acta Psychologica, 137,* 151-171.

Jansen, P., & Watter, S. (2008). SayWhen: An automated method for high-accuracy speech onset detection. *Behavior Research Methods, 40,* 744-751.

Katzberg, D., Belke, E., Wrede, B., Ernst, J., Berwe, Th., & Meyer, A. S. (2014). AUDIOMAX: A software using an automatic speech recognition system for fast and accurate temporal analyses of word onsets in spoken utterances. Poster presented at the *International Workshop on Language Production 2014*, Geneva.

Kessler, B., Treiman, R., & Mullennix, J. (2002). Phonetic biases in voice key response time measurements. *Journal of Memory and Language, 47,* 145-171.

Kisler, T., Reichel, U. D., Schiel, F., Draxler, Ch., Jackl, B., & Pörner, N. (2016). BAS Speech Science Web Services - an update of current developments. *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016),* Portorož, Slovenia, May 23-28, 2016.

Laubrock, J., & Kliegl, R. (2015). The eye-voice span during reading aloud. *Frontiers in Psychology, 6,* 1432.

Levelt, W. J. M. (1989). *Speaking: From Intention to Articulation*. Cambridge: MIT Press.

Levelt, W. J. M. (1999). Models of word production. *Trends in Cognitive Sciences, 3,* 223-232.

Levelt, W. J. M., Roelofs, A., & Meyer, A. S. (1999). A theory of lexical access in speech production. *Behavioral and Brain Sciences, 22,* 1-75.

Marklund, U., Marklund, E., Lacerda, F., & Schwarz, I.-C. (2015). Pause and utterance duration in child-directed speech in relation to child vocabulary size. *Journal of Child Language, 42*, 1158-1171.

McGraw, K. O., & Wong, S. P. (1996). Forming inferences about some intraclass correlation coefficients. *Psychological Methods, 1,* 30-46.

Metzing, C. & Brennan, S. E. (2003). When conceptual pacts are broken: Partner-specific effects in the comprehension of referring expressions. *Journal of Memory and Language, 49,* 201-213.

Mousikou, P., & Rastle, K. (2015). Lexical frequency effects on articulation: A comparison of picture naming and reading aloud. *Frontiers in Psychology, 6,* 1571.

Pechmann, T., Reetz, H., & Zerbst, D. (1989). Kritik einer Messmethode: Zur Ungenauigkeit von Voicekey Messungen [Critique on a measurement method: About the inaccuracy of voicekey measurements]. *Sprache & Kognition, 8,* 65-71.

Protopapas, A. (2007). CheckVocal: A program to facilitate checking the accuracy and response time of vocal responses from DMDX. *Behaviour Research Methods, 39,* 859-862.

Rabiner, L. (1989). A tutorial on hidden Markov models and selected applications and speech recognition. *Proceedings of the IEEE*, *77*, 257-286.

Rastle, K., & Davis, M. H. (2002). On the complexities of measuring naming. *Journal of Experimental Psychology: Human Perception and Performance, 28,* 307-314.

Rayner, K. (1998). Eye movements in reading and information processing: 20 years of research. *Psychological Bulletin, 124*, 372-422.

Rosenfelder, I., Fruehwald, J., Evanini, K., & Jiahong, Y. (2011). FAVE (Forced Alignment and Vowel Extraction) Program Suite. Retrieved from http://fave.ling.upenn.edu.

Roux, F., Armstrong, B. C., & Carreiras, M. (2016). Chronset: An automated tool for detecting speech onsets. *Behavior Research Methods.*

Sadat, J., Martin, C. D., Alario, F. X., & Costa, A. (2012). Characterizing the bilingual disadvantage in noun phrase production. *Journal of Psycholinguistic Research, 41,* 159-179.

Schiel, F. (1999). Automatic phonetic transcription of non-prompted speech. *International Congress of Phonetic Sciences 14,* 607-610.).

Schiel, F. (2015, November 5). Munich Automatic Segmentation. Retrieved from http://www.bas.uni-muenchen.de/Bas/BasMAUS.html.

Severens, E., van Lommel, S., Ratinckx, E., & Hartsuiker, R. J. (2005). Timed picture naming norms for 590 pictures in Dutch. *Acta Psychologica, 119,* 159-187.

Sichelschmidt, L., Jang, K.-W., Koesling, H., Ritter, H., & Weiß, P. (2010). Alignment in aufgabenorientierten Dialogen: ein multimodales Such- und Vergleichskorpus. [Alignment in task-oriented dialogues: A multimodal search and comparison corpus]. *Linguistische Berichte, 222,* 205-230.

Sjerps, M. J., & Meyer, A. S. (2015). Variation in dual-task performance reveals late initiation of speech planning in turn-taking. *Cognition, 136,* 304-324.

Strunk, J., Schiel, F., & Seifart, F. (2014). Untrained forced alignment of transcriptions and audio for language documentation corpora using WebMAUS. In N. Calzolari, K. Choukri, T. Declerck, H. Loftsson, B. Maegaard, J. Mariani, A. Moreno, J. Odijk, & S Piperidis (Eds.), *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC 2014)*, Reykjavik, Iceland, May 26-31, 2014.

Tyler, M. D., Tyler, L., & Burnham, D. K. (2005). The Delayed Trigger Voice Key: An improved analogue voice key for psycholinguistic research. *Behavior Research Methods, 37,* 139-147.

Young, S. (1996). A review of large-vocabulary continuous-speech recognition. *IEEE Signal Processing Magazine*, *13*, 45-56.