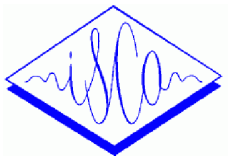

Third Workshop on Spoken Language
Technologies for Under-resourced Languages
Cape Town, South Africa
May 7-9, 2012

QUALITY MEASUREMENTS FOR MOBILE DATA COLLECTION IN THE DEVELOPING WORLD

Jaco Badenhorst, Alta de Waal and Febe de Wet

Human Language Technology Competency Area, CSIR Meraka Institute

ABSTRACT

The collection of speech data suitable for speech technology development is a challenge for under-resourced languages. Factors such as cost, availability of mother-tongue speakers and vast geographic distances call for techniques to optimise the data collection process in order to reduce re-collection of data. The use of mobile devices facilitate remote speech data collection. Although mobile (and remote) data collection addresses the challenging factors mentioned above, the environment is still less controlled than in the case of laboratory or studio-based recordings. In this paper we firstly revisit semi-realtime, basic quality control checks as implemented on available mobile-based speech data collection software (Woefzela). In addition, we introduce a quality control technique that uses speech duration estimation to validate the acoustic quality of the speech samples. We compare both techniques with manual verifications.

Index Terms— speech data collection, resource-scarce environment, under-resourced languages, automatic speech recognition, mobile data collection, android, Woefzela

1. INTRODUCTION

High quality, transcribed speech corpora have always been a cornerstone in the development of speech technologies. For under-resourced languages this requirement poses significant and unique challenges. The collection of new or additional speech data is usually the first step when speech technology is developed in these languages. Developing-world environments tend to complicate data collection, mainly because of additional limitations such as cost [1], a lack of general infrastructure (for example, reliable, cheap access to cloud computing resources [2]) and even the accessibility of mother-tongue speakers due to small, remote communities or widely distributed speaker populations.

Given this scenario, the availability of relatively inexpensive hand-held devices [3] presents exciting new opportunities for efficient speech data collection. For example, DataHound [4] is a commercial speech data collection application developed for the Android platform. It has been used successfully to build speech corpora using smart phones in the developed world. The DataHound approach has been extended to a new tool, Woefzela [5], that has been specifically adapted for use in the developing world.

Woefzela has enabled the recording of large numbers of geographically distributed speakers in areas with very limited infrastructure. For example, no internet access is required to collect data. However, data collection with Woefzela still required some training of the respondents. Although care was taken with regard to the recording environment, accidental and/or uncontrolled ambient noise and respondent errors still presented a real challenge. One way to improve on the quality of the speech data collection, is to capitalise on the limited processing power of current hand-held devices. For this reason, simple quality checks were implemented on Woefzela as semi-realtime processes. This strategy resulted in the cost-effective collection of more high quality speech data [5].

Basic quality checks do not identify all erroneous recordings, such as transcription errors, stuttering or intervening background speech. In this paper we introduce more advanced quality checks based on utterance length estimation. These checks aim to identify additional erroneous recordings, targeting transcription mismatches, that were not identified by the first on-device validation phase[1]. Both quality measurements are evaluated in terms of their ability to distinguish between useful and potentially corrupted or inappropriate recordings. Four corpora that were collected using Woefzela in developing world conditions are validated.

The next section gives an overview of related techniques and results that have been reported on in literature. The two validation techniques are subsequently described in Section 3. Section 4 describes the experimental set-up and results. A discussion and conclusions follow in Section 5.

2. BACKGROUND

Former studies on speech corpus validation mostly report on the off-line verification of previously collected data [6]. However, some of the best-practice proposed for off-line validation could also be implemented in semi-realtime validation on mobile devices. For instance, testing for empty files, average clipping rate, Signal-to-Noise Ratio (SNR), mean amplitude and file duration have proven to be effective practical acoustic quality measurements [6] that can be derived automatically from the acoustic data. Furthermore, speech file duration is especially useful to validate sound quality. (Extremely long or short file durations can indicate serious recording defects [6].) Another useful check to perform is to ensure that each file has a minimum period of silence at the beginning and/or end of the file. For speech corpora that consist of short utterances stored in separate files, acoustic measures should be computed over the complete file and the outcomes averaged over all the files of a speaker/session in order to spot corrupted sessions.

Automatic validation has the undisputed advantage that it is fast and provides a consistent level of precision. However, Van den Heuvel et al. [6] state that, as a general rule, content checks do need human intervention. On the other hand, experience has shown that it is hard to find a collection of acoustic features that, when evaluated automatically, equals the impression of human judgement [6]. These observations seem to indicate that not all aspects of automatic validation should be evaluated in terms of a human benchmark.

The first validation technique implemented in our approach applies simple quality checks based on signal energy measurements. The efficiency of this automatic quality check is determined by

[1] Although the second evaluation pass could potentially also run on a mobile device without interfering with the data collection process, this study is based on an off-line implementation.

its ability to distinguish between signal segments corresponding to voice activity and the pauses or silences between them, commonly referred to as voice activity detection (VAD). It is generally difficult to obtain an accurate indication of the presence or absence of speech, especially when the speech signal is corrupted by background noise or unwanted interference. Various features have been suggested to perform VAD, including short-time energy, zero-crossing rate, linear prediction coefficients, Cepstral coefficients, spectral entropy, least-square periodicity measure and wavelet transform coefficients [7]. However, despite the potential discriminating abilities other features may have, signal energy remains the basic feature that is used to perform VAD. Most standardised algorithms use energy in addition to other metrics to make a decision on the presence or absence of voice [7].

For these algorithms, during VAD, features are extracted from the input signal and compared with amplitude thresholds. Voice activity is detected if the measured values exceed the established decision thresholds for non-voice segments of training data. Decision thresholds therefore dictate the accuracy of the VAD algorithm. The added ability to adapt these thresholds according to time-varying changes in acoustic environments yield more reliable results. In this study we do not aim to improve on state of the art VAD algorithms, but rely on standard techniques that have been proven to be reliable and robust. We therefore use root mean square (RMS) signal energy and adapt the amplitude based decision thresholds, compensating for ambient acoustic energy.

The other automatic measure that we implement as part of our validation strategy is related to speech rate estimation and normalisation. It is known that the duration of individual phonemes varies for different speakers. In an analysis of phoneme durations in the TIMIT corpus, an eigenvector analysis of phone lengths showed that the first eigenvector contains more than 22% of the total information and approximately 65% is contained within the first 10 eigenvectors [8]. It was found that the first eigenvector corresponds to a simultaneous stretching of all phonemes (an indication of speaking rate). The second eigenvector corresponded to a differential lengthening of vowels in comparison with consonants, whereas the third seemed to indicate a distinction between the relative lengths of liquids, glides and nasals in comparison with plosives, fricatives and certain vowels [8]. On an utterance level, we expect that the differences of the second and third eigenvectors would largely cancel out, leaving an even greater percentage of variability that can be ascribed to speech rate differences for this level.

A technique is used in [9] to normalise local speech rate, given the local speech rate curve. In this manner, stretches of speech with fast local speech rate can be slowed down to the average speech rate and stretches of slow speech can be accelerated. According to [9], stretches of accelerated/decelerated speech exists more commonly at the utterance final and start positions for sentences. The statement is also made that speaker specific speech rate varies between as much as 70% and 140% for longer utterances consisting of whole sentences. Only a moderate correlation between local syllable rate and local phone rate was found ($r = 0.6$). Thus, the information content of phone durations differ from the information content of durations at the syllable level. The scope of the work we report on in this paper is limited to indicators of speech rate calculated at the phone level.

3. QUALITY CONTROL IMPLEMENTATION

De Vries et al. [5] introduced semi-real-time quality control (QC) methods that are all based on the calculation of RMS values given a waveform. These values are valuable indicators to characterise speech versus non-speech sounds for a piece of audio and, with careful implementation, require relatively low amounts of processing power. This approach was used to implement *QC-on-the-go* [5], i.e. simple recording quality checks (mainly volume control and start/stop error detection). In this paper, we extend this approach and provide a more sophisticated evaluation of the speech data leading to possible additional techniques. The new approach consist of two main elements: (1) Semi-realtime waveform analysis for basic quality evaluation and (2) Speech duration estimates for speech sufficiency evaluation. We analyse the collected speech data of four languages, applying the above techniques to judge quality and present our findings.

3.1. Semi-realtime waveform analysis for basic quality evaluation

To characterise and detect the separate elements (speech and non-speech) of a single recording, direct analysis of the waveform is performed. We call this method the 'sliding window' principle. Only a fixed number of samples are considered at a time (window), allowing a direct comparison with all other adjacent energy segments in time. For this purpose we define a window segment of size $w = 0.05$ seconds, which is generally about half the duration of a single vowel sound. Choosing a segment of this size ensures the presence of adequate voicing for speech detection, given any particular window. Segment size also places limits on the granularity of the speech detection that can be performed. A longer window size effectively implies averaging over more samples, which leads to reduced sensitivity of the measurements taken. The chosen segment duration provides a good operating point in terms of this trade-off.

To progress through an entire recording, we introduce a specific window step-size that defines the starting index of the next window. Choosing a step size that is smaller than the window size is desirable, since this also provides the required resolution (smoothing) for the measurement. Having a much smaller step size than the segment size allows for the detection of gradual changes in amplitude, effectively representing a smaller time scale. Consequently, we choose a value of $t = 0.005$ seconds as the step-size parameter. This allows for 10 measurements to be taken for any particular time-period and is roughly comparable to the duration of short vowel sounds.

We calculate signal energy for the samples i of every window, by determining the root mean square (RMS_{window}) value as follows:

$$RMS_{window} = \sqrt{\frac{1}{n}\sum_{i=1}^{n} S_i^2} \quad (1)$$

where S_i^2 is the square value of a specific sample.

Using these RMS_{window} measurements it is then possible to effectively classify each window as a speech or a non-speech segment. RMS_{window} value amplitude plots (Figure 1 is an example) can be drawn using all of the calculated RMS segment values for a specific recording.

Sample utterances can be recorded with specific recording conditions in mind. We collected samples simulating various ambient conditions, for example: recording in a kitchen with the kettle boiling, air-condition running, outside (wind blowing, bird twitter etc.) and background speech. From the amplitude plots of these utterances it was possible to observe the effect that each of the noise and speech sources (which affect recording quality) has on the RMS_{window} amplitudes. These initial trails clearly showed the amplitude in speech sounds to be much greater than most other background noise sources (with the exception of background speech).

For good recording conditions (indoors and relatively silent environment) it is even fairly easy to distinguish between a fricative phone and the background RMS_{window} values.

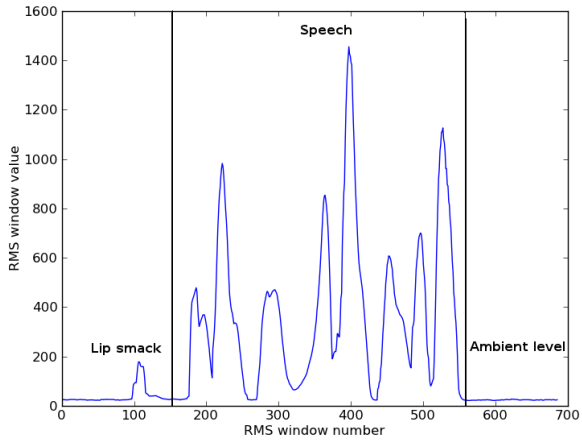

Fig. 1. Example of RMS window amplitudes (RMS_{window}) for a single recording, with window segment of size $w = 0.05$ and step-size $t = 0.005$ seconds.

3.1.1. Sensitivity and silence detection

Channel or background noise is seen as an additive amplitude for the RMS_{window} values (as calculated in Equation 1). Consequently, any amplitude thresholds that can be used are subject to these ambient noise levels, as well as the sensitivity of the microphone of the specific recording device. To deal with the ambient noise levels in our experiments, we estimate the mean ambient RMS level ($\mu_{ambient}$) for a batch of recordings.

Specifically, for a given recording, we accumulate the lowest RMS_{window} amplitude values. A fixed number ($n = 20$) of windows (the combined data samples are roughly comparable to the duration of a single long vowel sound) yield sufficient data to characterise the background amplitude for a given recording. Accumulating these values for all recordings of a session finally give the general ambient noise statistic for a particular session ($\mu_{ambient}$).

Silence detection is performed by marking all windows with an RMS_{window} amplitude below a specific threshold (sil) as silence. After background ambient levels have been taken into account (see section 4.2), the non-speech duration (D_{sil}) of a recording can then be calculated as $D_{sil} = ts$, where t is the window step-size and s the number of windows marked as non-speech.

3.1.2. Speech corruption measurements

We present three basic quality checks, based on mandatory rules for making a valid recording. These are (1) clipping detection, (2) volume detection and (3) speech cutting detection.

Clipping of the waveform occurs when the input signal on the recording device is too big. This saturation can occur at the microphone or at any stage of signal processing. In order to detect signal saturation, we analyse the audio samples of a particular segment (window). Finding a single clipped sample amplitude will result in the utterance being marked as a clipped recording.

Volume detection is aimed at determining whether an utterance contains any speech, with a chance for good signal-to-noise ratio (SNR). We set the vol threshold (as described in 4.3). If the RMS_{window} value of any window in the utterance is found to be greater than this value, the utterance is considered as a speech recording.

It may be that a recording contains speech that is cut off right at the beginning or end. This will happen for example if the respondent started talking before pressing the record button. To detect such recordings the RMS_{window} values for the first and last $t = 0.025$ seconds are considered. If any of these RMS_{window} values are found to have a value greater than the cut threshold, the recording is flagged as an occurrence of speech cutting.

3.2. Speech duration estimates for speech sufficiency evaluation

Given the transcriptions of the utterances to be recorded during data collection, it is possible to derive an estimate for the expected length (D_{exp}) of any particular prompt. Dividing the representation of a text prompt into sub-units, such as phonemes, allow prompt durations to be estimated from a relatively small set of sub-unit durations. However, achieving low variability for this measure requires another crucial component: speech rate. For short utterances, every speaker generally speaks more or less at the same rate, but these rates may still vary significantly between speakers.

Speech duration estimates therefore rely on *speech rate measurements* as well as *sub-unit estimation*. In this study, we present two possible sub-unit estimation techniques. The first relies on average phone lengths from a surrogate speaker, while the second is a data-driven approach (section 3.2.2). Speech rate measurements as well as the sub-unit estimation techniques are subsequently used to derive verdicts regarding the observed duration of speech utterances.

3.2.1. Speech rate measurement

A simple way to measure speech rate is the average number of phones per second. It is also possible to measure speech rate indirectly, as a relative value given a reference speaker. For a number of n recordings one can then estimate the general speech rate mismatch α as a scaling value to the reference speaker's speech rate. Using silence detection (section 3.1.1) to obtain the value D_{sil} and the total duration of recording D_{tot}, we calculate the seen speech duration for a specific recording: $D_{det} = D_{tot} - D_{sil}$. For the same bundle of recordings (n), the mean speech rate mismatch α can then be calculated as

$$\alpha = \frac{\sum_{i=1}^{n} D_{det}}{\sum_{i=1}^{n} D_{exp}} \quad (2)$$

where the ratio of the mean detected D_{det} and expected D_{exp} prompt durations is determined.

3.2.2. Sub-unit estimation

To obtain a value for D_{exp}, we utilise sub-unit durations and compare two different sources for these units. These are:

- the average phone lengths from a surrogate speaker (TTS technique) and;
- a data driven approach to sub-unit estimation ($Data$ technique).

Both approaches currently use grapheme-to-phoneme (G2P) pronunciation prediction to identify sub-unit classes for required text prompts. This means that for every unique phone-label generated during the prediction process, a sub-unit duration class must exist.

The TTS technique relies on annotated, high quality audio data (preferably of the same speaking style) to estimate these lengths. A mean duration of every phone from the surrogate speaker is estimated. We utilised audio data used for the construction of TTS systems (that were available to us - section 4.1) for this purpose. Next, we sum these durations for the predicted G2P pronunciation string to obtain an initial expected duration for the specific text prompt. Lastly, we also scale this surrogate-specific expected prompt duration by the estimated speech rate mismatch α (Equation 2) to obtain the final expected prompt duration D_{exp}.

With the data driven approach ($Data$ technique) the goal is to learn 'relative' durations for sub-classes that, if used together to predict prompt durations, yield good estimates for the speech duration D_{exp} of any given recording of the text corpus. For this analysis, similar to the surrogate speaker method, phone labels are used as the sub-classes.

Our algorithm requires n recorded prompts of the same speaker to firstly estimate relative durations for each of the sub-units. The algorithm starts off by initialising every sub-unit with the mean class duration (μ_{class}) and a seen example count of $C_{class} = 1$. This value can be obtained using all of the detected prompt durations, D_{det} and matching G2P pronunciation sequences across the different speakers for the dataset of a particular language.

Sub-unit duration is estimated by iteratively computing the expected prompt duration, D_{exp} (similar to the surrogate speaker approach) and then finding the prompt specific speech rate mismatch α_{rec} (Equation 2). For each of the sub-units of the current recording we then compute an updated unit example ($Sub_{update} = \alpha_{rec} * \mu_{class}$). Finally, each of the seen sub-unit mean durations (μ_{class}) is adjusted using the recurrence relation:

$$\mu_{class} \quad = \quad \mu_{class} + \frac{Sub_{update} - \mu_{class}}{C_{class}} \tag{3}$$

where μ_{class} is the current sub-unit specific class mean, Sub_{update} the scaled sub-unit duration example and C_{class} the class specific example count. C_{class} is incremented for every example Sub_{update}.

The final speaker specific expected prompt duration D_{exp} can then be determined for the current recording, given n previous recordings of the same speaker. We simply sum the obtained sub-unit durations according to the G2P prediction sequence.

3.2.3. Speech sufficiency evaluation

The main purpose of speech sufficiency evaluation as a quality measurement is to deduce (with some certainty) whether the quantity (in time) of recorded speech in an utterance matches the duration of speech that can be expected given its transcription. When the expected (D_{exp}) and detected durations (D_{det}) of speech is seen to be within an acceptable region, the recording is marked to be of good quality.

To successfully evaluate prompt durations, our algorithm has to deal with two types of variability. These are: 1) general speech rate estimation error and 2) intra-speaker variation, respectively. General speech rate estimation error is a function of the amount of quality speech data available for the specific speaker, while intra-speaker variation captures all other deviations from the prompt level speech rate of a speaker. Possible causes of such deviations include: variable local speech rates, a particular language and mean prompt length in words.

We therefore define the size of the acceptable region (λ_{accept}) as:

$$\lambda_{accept} \quad = \quad \beta(\sigma_\alpha + \sigma_{intra}) \tag{4}$$

where σ_α and σ_{intra} represent the uncertainty in seconds that is associated with speech rate estimation error and intra-speaker variability, respectively.

We use the algorithms as proposed in section 3.2.2 to estimate a value for α and therefore σ_α, while the values for σ_{intra} are obtained from data analysis of specific speakers (section 4.5).

4. EXPERIMENTAL SET-UP

4.1. Speech data

The Department of Arts and Culture (DAC) of the South African government is sponsoring a large-scale project to develop speech resources for all eleven official South African languages. The first milestone of the project is to develop state-of-the-art tools and techniques for speech resource acquisition, annotation and verification, while concurrently developing samples of the speech resources to be delivered for all eleven languages. In particular, $50 - 60$ hours of broadband speech data were collected per language. Typically, a recording session consists of 500 recordings from a mother-tongue speaker of the respective language. A recording is an utterance of ± 3 words. For the purpose of this research, we select recording sessions with a minimum of 350 recordings per session, yielding 181 Afrikaans (Afr), 201 English (Eng), 194 Sepedi (Nso) and 158 isiZulu (Zul) speaker sessions. We use these sets of data for all subsequent experiments.

The TTS data came from systems that were developed for all the official languages of South Africa and the resources can be found at http://www.meraka.org.za/lwazi. Here, a single speaker (per language) recorded 300 utterances of length ± 10 words. These recordings are specifically made with phone coverage in mind and usually employs a constant speaking style. Moreover, TTS data often includes accurate phone alignments (as this is very important when building a TTS voice). Such a dataset typically provides more data of a single speaker than is generally the case for automatic speech recognition (ASR) corpora.

4.2. Sensitivity measurement

As described in section 3.1.1, background ambient levels and the recording device influence RMS amplitude levels. For RMS amplitudes this is an additive component that can be compensated for in a linear fashion. The experiments in this paper rely on the mean ambient RMS level ($\mu_{ambient}$) to compensate for session specific ambient levels. Figure 2 shows $\mu_{ambient}$ for the collected speech data (section 4.1). Ordering from low to higher ambient levels indicates that most of the session ambient energy are at acceptable levels ($\mu_{ambient} < 100$). For the few sessions with values $\mu_{ambient} > 100$, it is conceivable that lower signal to noise ratios could be expected.

We perform sensitivity adjustment on all silence detection thresholds (sil) to improve robustness of the duration estimation technique. This is accomplished through addition of the session specific ambient noise level ($\mu_{ambient}$) to the chosen silence detection threshold sil.

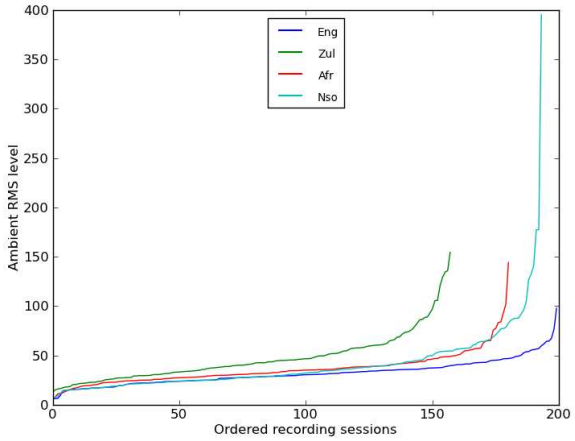

Fig. 2. Ordered $\mu_{ambient}$ levels for the recording sessions of different languages.

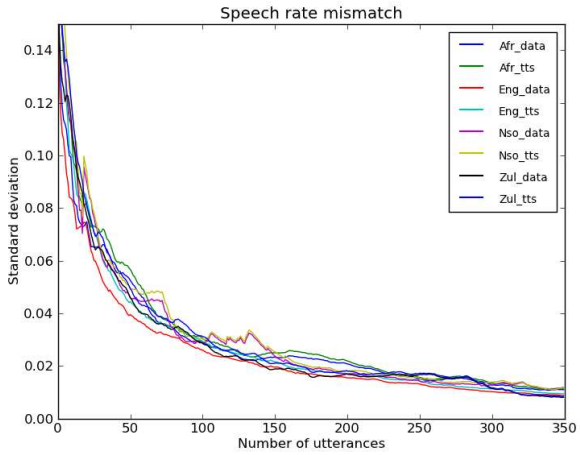

Fig. 3. General speech rate confidence σ_α for different sub-unit estimation techniques (*TTS* and *Data*) and four languages.

4.3. RMS window thresholds

Various RMS amplitude threshold values are chosen to evaluate specific waveform characteristics in sections 3.1.1 and 3.1.2. For audio recordings using the WAV format and PCM encoding, 1 channel at 16 bit sample width and a sampling frequency of 16 kHz, these can be described by the following parameters:

- Silence threshold $sil <= 100$
- Volume sufficient threshold $vol >= 600$
- Speech cutting threshold $cut >= 300$

These thresholds were determined by inspection (given the same recording devices that were used for data collection). The silence threshold value, with sensitivity adjustment required for higher ambient levels (4.2), was derived from recordings made in relatively quiet environments. A volume threshold is set to indicate whether at least some speech are captured. Lastly, the speech cutting threshold that is used to flag recordings with a higher RMS_{window} value for any window during the time periods ($t = 0.025$) specified in Section 3.1.2, is set.

4.4. Speech rate estimation accuracy

The algorithms in section 3.2.2 can be used to estimate speech rate with specific confidence (σ_α). A general estimate for this value can be derived given a large number of recordings. To investigate σ_α, we incrementally measure the standard deviation of the general speech rate mismatch (Equation 2). Every iteration then reflects a value for a different fixed number of recordings across all speakers (with an equal contribution per speaker) of a particular language. Figure 3 shows a plot describing the parameter σ_α given a specific number of recordings.

When incorporating average phone lengths form a surrogate speaker, the above operation also requires one additional step of normalisation, since the iteration specific result, α_x, is still relative to the surrogate speaker. Dividing α_x by α_{max}, the best estimate, provides values for α that are centred around unity.

Speech rate ω can also be represented in terms of phones per second for the fixed number of recordings x. Specifically $\omega_x = \frac{n_p}{D_{tot}}$, with n_p the number of phoneme examples. Normalising by ω_{max} for the specific speaker is a comparable measurement to σ_α.

	Afrikaans	English	Sepedi	isiZulu
σ	0.198	0.155	0.215	0.274
2σ	0.377	0.310	0.429	0.548
3σ	0.565	0.465	0.644	0.822

Table 1. Alignment of σ_{intra} for the TTS speakers of different languages.

4.5. Intra-speaker variance

For this analysis, intra-speaker variation serves to quantify all seen variation for the speech duration estimate after speech rate normalisation. Working in a resource-scarce environment implicates that very little verified transcribed and aligned speech data may be available. One resource for the languages Afrikaans, English, Sepedi and isiZulu is speech data from a TTS system (section 4.1).

To analyse intra-speaker variance we divide the TTS recordings into chunks of comparable size (1-3 words) to the newly collected sets of data we analyse. σ_{intra} can then be calculated for n recordings as:

$$\sigma_{intra} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(D_{exp_i} - D_{det_i} - \mu_\Delta)^2} \qquad (5)$$

where $\mu_\Delta = \frac{1}{n}\sum_{i=1}^{n}(D_{exp_i} - D_{det_i})$, D_{exp_i} is the expected duration and D_{det_i} the seen speech duration for recording i.

Finally, the estimated alignment σ_{intra} for the expected D_{exp} and detected D_{det} durations of the four TTS speakers (one for every language) is given in Table 1. It is clear that this value differs significantly between languages, indicating that it is probably language dependent. Transcriptions for Sepedi and isiZulu in general give longer expected durations than for Afrikaans and English (which are similar in this respect) and may be an important factor to explain the differences.

It is also possible to analyse the σ_{intra} of the collected data. Ordering speaker sessions with regard to ambient levels $\mu_{ambient}$

Sessions	Afrikaans	English	Sepedi	isiZulu
50	0.733	0.582	0.682	0.832
100	0.712	0.588	0.696	0.821
150	0.687	0.579	0.802	0.861

Table 2. Estimation of $\mu_{\sigma_{intra}}$ for the collected data (section 4.1) when ordering according to ambient background levels.

Criteria	Low Vol	Cut	Speech Suff
Speech is not part of the recording.	x		
Speech cannot be interpreted due to soft volume.	x		
Volume is low compared to the other recordings in the test.	x		
Speech has been cut while being uttered.		x	
Reverberation (echo) due to the same speaker can be heard		x	x
Consistent background noise present (ex. wind blowing).		x	x
A definite isolated noise event at the start or end of the recording.		x	x
The transcription and the speech in the recording is not a 100% fit.			x
Stuttering and/or hesitations.			x
Acronym (spelled out letters) - A letter or a group of letters are sounded out and/or not a known word (such as email, mweb)			x

Table 3. Criteria for manual verification experiments.

and taking the mean iteratively we obtain the values in Table 2. It is clear that the measured values are much larger for the 'dirty' data of all languages and smaller σ_{intra} values can be expected for the subset of high quality data.

4.6. Manual verification

We designed manual verification experiments for the three languages Afrikaans, English and Sepedi. Each experiment involved three respondents and we used the verdict with the majority vote (2 out of 3). Each respondent verified two sets of recordings (one for each verification experiment). Verification itself was divided into two main categories, i.e. the basic quality checks described in section 3.1.2 and the speech sufficiency measure. The recordings were drawn at random, ensuring 50% male and female speaker division. Table 3 lists the criteria and the corresponding tests as it was used in the manual verification experiment.

Only test volume and speech cutting were verified (with two independent experiments) as basic quality checks, since clipping can accurately be detected automatically. The 100 test items for these experiments were drawn only from the set of recordings that did not pass the relevant QC test.

The 50 recordings identified for the speech sufficiency verification experiment "passed" all three basic quality checks. These recordings were chosen from the data that did not pass the speech sufficiency test, since this work are mainly concerned with the correct identification of unacceptable recordings. Experiments were subsequently conducted to evaluate the acceptable region as defined by Equation 4. Values for β are chosen to be 3 or 2, respectively to allow evaluation of two different boundaries. We choose the value $\sigma_\alpha = 0.02$, which is well above the value for σ_α at 350 recordings in Figure 3. The second component, σ_{intra}, are the values from Table 1.

4.7. Results

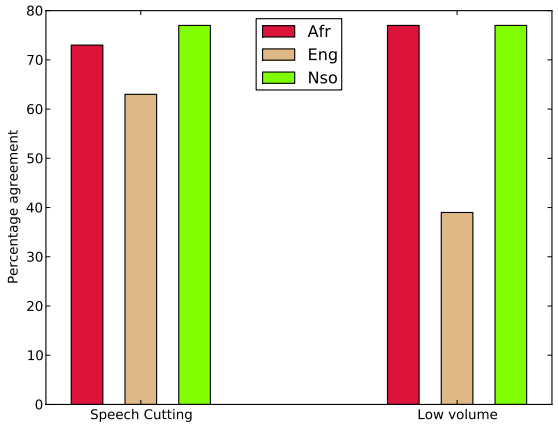

Fig. 4. Human verification: Basic quality checks.

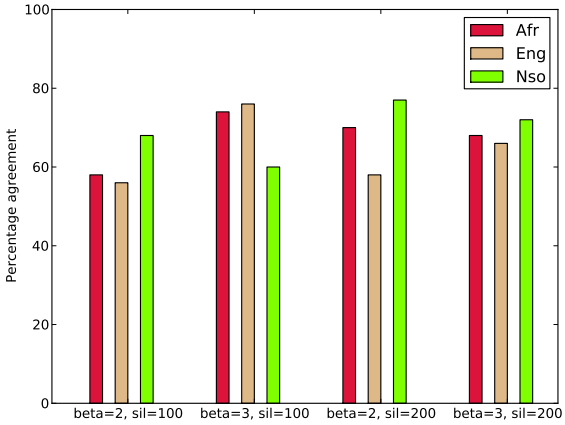

Fig. 5. Human verification: Speech sufficiency measure.

Figures 4 and 5 present the results from the manual verification tests for Afrikaans, English and Sepedi. The bar graphs indicate the agreement between manual and automatic verdicts. For example, the first red bar from the left in Figure 4 can be interpreted as follows: 73% (73 from 100 samples) of the manual verdicts agreed with the automatic verdicts that speech was cut for Afrikaans.

Figure 4 represents the agreements of basic quality checks and Figure 5 the agreements of speech sufficiency. Only the verdicts regarding unacceptable data are indicated in Figure 5. For both Afrikaans and Sepedi, human agreement on the automatic verdicts is higher than 60% (in some cases almost 80%) for the unacceptable data samples. Agreement for low volume recordings in English is the exception and this can possibly be explained by the low ambient energy conditions as is also reflected in Figure 2.

Interestingly, Figure 5 indicates that an RMS amplitude threshold for silence of 100 gives better performance for $\beta = 3$, but with a threshold of $sil < 200$, $\beta = 2$ is best. It may be that high ambient noises are detected better as $\beta = 3$ errors, given this sensitivity. An exception to this rule is Sepedi, where higher ambient levels in general may play a role.

The work done in this article focuses on the speech duration estimation technique in order to determine speech sufficiency of a speech sample. Basic quality checks are also put in place as a mechanism to filter unacceptable speech before speech sufficiency is tested. Figure 6 illustrates the overlap of techniques described (for the full set of data - section 4.1) and this indicates the individual gain of each technique. The overlap is measured between basic quality check techniques and speech duration estimation and not between individual basic quality check techniques (clipping, cutting and volume). The coloured bars represent the number of recordings deemed unacceptable by the technique itself. In contrast, the black bars show the number of recordings deemed unacceptable, using the other technique. The delta represents the gain from the technique in isolation and it can be seen from this graph that the speech sufficiency measure contributes significantly (more than 25%) to the set of unacceptable recordings.

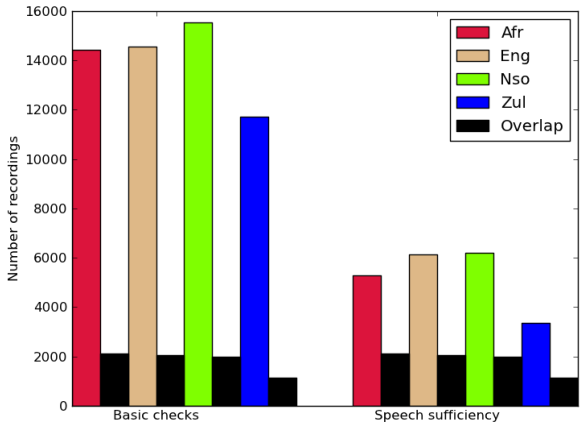

Fig. 6. Comparing basic quality evaluation and speech duration estimate techniques for different languages.

5. DISCUSSION AND CONCLUSIONS

This paper presents a series of quality measurements for speech data collection using hand-held devices. With careful implementation, these techniques could potentially benefit the collection effort by improving the efficiency of the recording process. A sliding window technique is implemented [5], defining an RMS amplitude measurement for a chosen fixed segment size that allows adequate detection of voicing for the speech signal. Detection of voicing utilises amplitude thresholds. We implement adaptive amplitude thresholds. This improves the speech sufficiency technique by compensating for ambient levels, since these effects are seen as additive amplitude.

Since the average speech rate of different speakers vary quite significantly, the speech sufficiency technique with regard to duration estimates requires speech rate adjustment for accurate speaker specific measurements. Two techniques are presented to accomplish speech rate adjustment. We show that they exhibit similar accuracy. In fact, speech rate can be determined quite accurately in general, even from a small number of recordings (20-50). Furthermore, this accuracy is doubled at 200 recordings where the contribution with regard to variability of this component is effectively an order less than intra-speaker variability.

Estimating intra-speaker variance using TTS data shows relatively good alignment. Comparatively, we show that this value for sessions with less stringent recording conditions (containing some errors) differ significantly. It is almost three times the TTS estimate. Manual verification confirms that using the boundaries $\beta = 2$ and $\beta = 3$ for the acceptable region (λ_{accept}) is indeed a fruitful strategy. Assuming the normal distribution we expect the actual values of intra-speaker variance (for the collected data) to be lower, since there are about 70% agreement on unacceptable verdicts for $\beta = 3$. At the $\beta = 3$ boundary of the acceptable region (λ_{accept}), we show that the basic quality checks and the speech sufficiency technique have low overlap and indeed identify different types of error. This promising result strengthens the argument for inclusion of the speech sufficiency technique in mobile data collection. Applying both techniques during data collection should therefore ensure that data of acceptable quality is captured and data re-collection is avoided.

6. REFERENCES

[1] C. van Heerden E. Barnard, J. Schalkwyk and P.J. Moreno, "Voice search for development," in *Proc. Interspeech*, Makuhari, Japan, September 2010, pp. 282–285.

[2] S. Horrigan M. Kam F. Metze A. Kumar, A. Tewari and J. Canny, "Rethinking speech recognition on mobile devices," in *Proc. 2nd International Workshop on Intelligent User Interfaces for Developing Regions*, Palo Alto, CA, February 2011, pp. 10–15.

[3] R. Kabir A. Park T. Hazen, E. Weinstein and B. Heisele, "Multimodal face and speaker identification on a handheld," in *Proc. Workshop on Multimodal User Authentication*, Santa Barbara, CA, December 2003, pp. 113–120.

[4] T. Hughes, K. Nakajima, L. Ha, A. Vasu, P. Moreno, and M. LeBeau, "Building transcribed speech corpora quickly and cheaply for many languages," in *Proc. INTERSPEECH*, Makuhari, Japan, September 2010, pp. 1914–1917.

[5] Nic J. de Vries, Jaco Badenhorst, Marelie H. Davel, Etienne Barnard, and Alta de Waal, "Woefzela - An open-source platform for ASR data collection in the developing world," in *Proc. INTERSPEECH*, Florence, Italy, August 2011, pp. 3177–3180.

[6] H. van den Heuvel, D. Iskra, E. Sanders, and F. de Vriend, "Validation of spoken language resources: an overview of basic aspects," *Language Resources and Evaluation*, vol. 42, pp. 41–73, 2008.

[7] K. Sakhnov, "Dynamic energy-based speech/silence detector for speech enhancement applications," in *Proc. World Congress on Engineering*, Hong Kong, July 2009, vol. 1, pp. 801–806.

[8] C.J. van Heerden and E. Barnard, "Speaker-specific variability of phoneme durations," *South African Computer Journal*, vol. 40, pp. 44–50, 2008.

[9] H.R. Pfitzinger, "Intrinsic phone durations are speaker-specific," in *Proc. ICSLP*, September 2002, vol. 1, pp. 1113–1116.