

# Fuzzy Recall and Precision for Speech Segmentation Evaluation

*Bartosz Ziółko\**, *Suresh Manandhar\**, *Richard C. Wilson\**

\*Department of Computer Science, University of York  
Heslington, YO10 5DD, York, UK  
{bziolko,suresh,wilson}@cs.york.ac.uk

## Abstract

Automatic evaluation of speech segmentation is problematic as predicted segment boundaries never align precisely. For this reason, most researchers apply ad-hoc methods for measuring the accuracy of speech segmentation. This makes judging the relative merits of each method extremely subjective and difficult. We address problem by proposing a new method for speech segmentation based on fuzzy logic. Our methods extend the well-known precision and recall metrics to the fuzzy case. The proposed method can be easily generalised to arbitrary sequence alignment and prediction problems and to arbitrary fuzzy similarity functions.

## 1. Introduction

Segmentation is a task of splitting a sequence into meaningful units. For a speech input, these units can be phoneme boundaries, word boundaries, sentence boundaries or turn-taking boundaries etc. The segmentation problem can be viewed as an unlabelled splitting problem where the input sequence needs to be split into a sub-sequences. Methods for evaluating the accuracy of the segmentation problem tend to be ad-hoc with researchers using their own methods such as allowing 10% overlap between the predicted segmented and the ground truth. For phoneme segmentation and other speech segmentation problems predicted segment boundaries never align accurately to the actual segment boundary. In addition, some segments are missing and extra ones are predicted. In this paper, we develop a systematic method for evaluating the accuracy of the predicted segmentation based on precision and recall (van Rijsbergen, 1979). Our method takes into account both the match between the number of segments predicted and the closeness between the predicted and the actual segment boundaries.

In the vast majority of approaches to speech recognition, the speech signals need to be divided into segments before recognition can take place. The properties of the signal contained in each segment are then assumed to be constant, or in other words to be characteristic of a single part of speech. Other levels of speech segmentation are often conducted as a part of further analysis in speech recognition systems.

In the next section we describe the problem of speech segmentation. We list examples of evaluation methods used by some authors in their research. We point out their flaws and lack of standardization. Then we give a short introduction to fuzzy logic and an explanation why it is useful. We present the detailed algorithm of evaluation method for phoneme speech segmentation. Finally we give an exemplar comparison of the suggested method with commonly used ones.

## 2. Speech segmentation

Speech segmentation can also be used for a number of different tasks (Fig. 1). Often it is used for word segmentation. This can be done by Viterbi and forward-backward

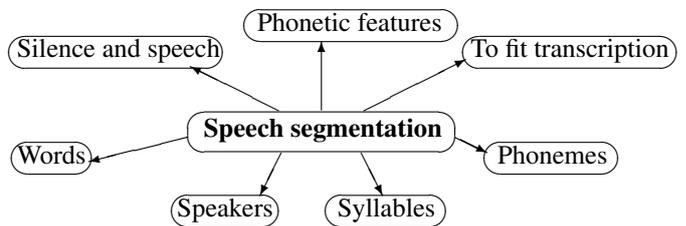


Figure 1: The types of speech segmentation.

segmentation (Demuynck and Laureys, 2002). Another applied method (Subramanya et al., 2005) is based on mean and variance of spectral entropy. A different problem covered by the same name is separating silence and speech from an audio recording (Zheng and Yan, 2004). The method uses so called TRAPS-based segmentation and Gaussian mixture based segmentation. Segmentation here means mainly removing non-speech events and additionally clustering according to speaker identities, environmental and channel conditions. Yet another possible segmentation is by phonetic features (not necessarily phonemes) (Tan et al., 1994), by applying wavelet analysis. There is also research on syllable segmentation (Villing et al., 2004). Another meaning is segmenting due to partially correct transcriptions (Cardinal et al., ). In this case segmentation is combined with recognition. We can also understand segmentation as the process of breaking audio into phonemes (Grayden and Scordilis, 1994; Ziółko et al., 2006).

In many applications, like speech recognition, the most common segmentation method is to use constant-time framing, for example into 25 ms blocks (Young, 1996). This method benefits from simplicity of implementation and the ease of comparing blocks of the same length. However, the different length of phonemes is a natural phenomenon which cannot be ignored. Moreover, boundary effects provide additional distortion. Constant segmentation therefore risks losing information about the phonemes due to merging different sounds into single blocks, losing phoneme length information and losing complexity of individual phonemes. A more satisfactory approach is an attempt to find the phoneme boundaries. A number of

approaches have been previously suggested for this task. Segmentation can be conducted by filter bank energy contours analysis (Grayden and Scordilis, 1994). Neural networks (Suh and Lee, 1996) have also been tested, but they require time consuming training. Segmentation can be applied by the segment models instead of the hidden Markov models (HMM) (Ostendorf et al., 1996). Partitioning is based upon the model decode. It allows boundaries to be located only on several fixed positions dependent on framing (on multiplied length of one frame). The analysis of the first derivative of power in different frequency subbands gives another opportunity to distinguish phoneme boundaries (Ziółko et al., 2006). Many phonemes exhibit rapid changes in particular subbands which can determine their beginnings and endpoints. The typical approach to phoneme segmentation for creating speech corpora is to apply dynamic programming for time alignment (Holmes, 2001). This method is very accurate but demands transcription and hand segmentation of some utterances to start with.

There are several types of speech segmentation and several approaches to most of them. Surprisingly evaluation methods for speech segmentation are quite simple and do not consider all situations. They are usually methods developed for a given solution, that are not very universal and lose accuracy in their simplification. Typically they are based on counting the number of insertions, deletions and substitutions of the automatic segmentations with respect to a hand-checked reference transcription. The automatic word segmentation (Demuynck and Laureys, 2002) was evaluated by counting the number of boundaries for which the deviation between automatic and manual segmentation exceeded thresholds of 35, 70 and 100 ms. The syllable segmentation (Villing et al., 2004) was evaluated by counting the number of insertion and deletion errors within a tolerance of 50 ms before and after a reference boundary. Some authors do not give any details about such tolerances or do not give such a tolerance but use generally the same method (Grayden and Scordilis, 1994). This insertion and deletion approach has a few flaws. First of all, a value of tolerance is questionable and, generally, can not be set on any theoretical bases. It is rather chosen empirically, quite often from experience in results of a given speech segmentation method. Moreover, such methods treat different inaccuracies as simply correct or wrong detections (or giving a larger scale of grades) without considering "how wrong" the detection really is. A tolerance is set to be 50 ms in example (Villing et al., 2004) for syllables, according to a statistically average length of a segment. The disadvantage of such approach is that speech segments, whatever they are, words, syllables or phonemes, vary in their length quite considerably. For this reason a shift of 50 ms in boundary location is not the same for a 100 ms long syllable as for a 300 ms long one. Our approach to speech segmentation evaluation considers those problems and try to include them in the evaluation result.

### 3. Fuzzy sets for recall and precision

Fuzzy logic is a tool for embedding structured human knowledge into workable algorithms. In a narrow sense,

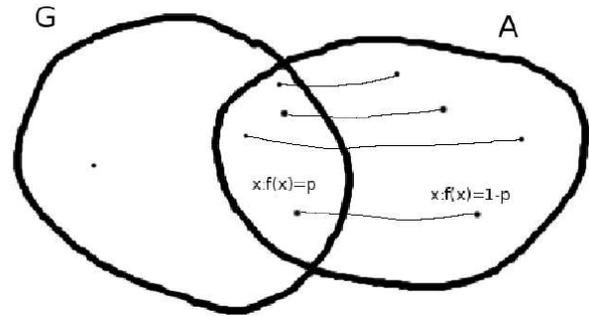


Figure 2: The general scheme of sets G with correct boundaries and A with detected ones. Elements of set A have a grade  $f(x)$  standing for probability of being a correct boundary. In a set G there can be elements which were not detected (in the very left part of the set).

fuzzy logic is considered a logical system aimed at providing a model for modes of human reasoning that are approximate rather than exact. In a wider sense, it is treated as a fuzzy set theory of classes with unsharp boundaries (Kecman, 2001). Fuzzy logic found many applications in artificial intelligence due to the introduction of the opportunity of numerical and symbolic processing of a human-like knowledge. This kind of processing is needed in proper evaluating of many types of segmentation. In our case we are interested in speech boundary (i.e. phonemes) location (fig. 3). Detected boundaries may be shifted more or less with respect to a manual segmentation. This 'more or less' makes a crucial difference and cannot be mathematically described in a Boolean logic. Fuzzy logic introduces an opportunity of grading detected boundary locations in more sensitive and human-like way.

Our segmentation evaluation method is based on the well-known recall and precision evaluation method. However, in our approach, calculated boundary locations are elements of a fuzzy set and a binary operation  $T$ -norm describes their memberships.  $T$ -norm is defined as a function  $T : [0, 1] \times [0, 1] \rightarrow [0, 1]$  which satisfies commutativity, monotonicity, associativity and for which 1 acts as an identity element. As usual in recall and precision, one set contains relevant elements. The other is the set of retrieved boundaries. We calculate an evaluation grade using the number of elements in each of them and in their intersection. The comparison of the number of relevant boundaries and a number of elements in intersection gives precision. In a boolean version of the evaluation method it is information about how many correct boundaries were found. By using fuzzy logic we evaluate not only how many boundaries were detected but how accurately they were detected. The comparison of the number of retrieved elements and intersection gives recall, which is a grade of wrong detections. In this case fuzzy logic allows to evaluate not only a number of wrong detections but also their incorrectness. Each retrieved boundary has a probability factor which represents being correct information.

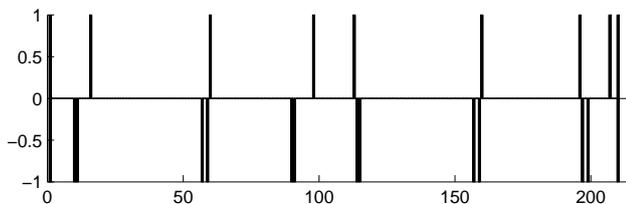


Figure 3: The example of phoneme segmentation of a single word. In the lower part hand segmentation is drawn. Boundaries are represented by two indexes close to each other (sometimes overlapping). Upper columns present the example of segmentation for the word done by a segmentation algorithm. All of calculated boundaries are quite accurate but never perfect

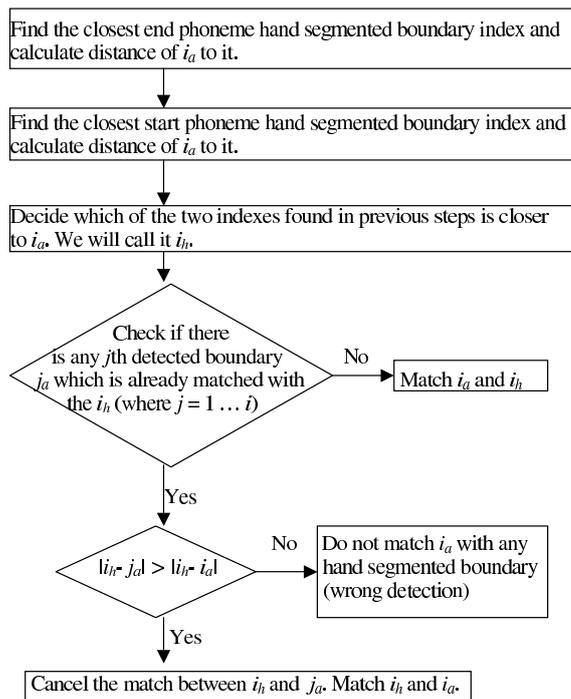
#### 4. An algorithm of speech segmentation evaluation

In this section we present an example of applying the approach described in the previous section for phoneme speech segmentation (fig. 3). Due to described features, such segmentation and its evaluation is particularly useful in speech recognition. In this case we have to make three assumptions:

- Hand segmentation is presented as a set of narrow ranges. Neighboring phonemes overlap each other in these ranges.
- Detected boundaries are represented as a set of single indexes.
- We assume the perfect detection of silence. Silence segments may be of almost any length. Due to this fact including them in evaluation would cause serious inaccuracy. This is why we skip silence segments in evaluation.

The method proceeds as follows:

1. Assign first and last detected boundaries with the same value as hand segmented boundaries (typically the first and the last index). This is done because of the third assumption.
2. Start with matching the closest detected and hand segmented boundaries. We need to match them in pairs. Each boundary may have only one matched boundary from the other set. Do following steps for each  $i$ th detected boundary  $i_a$  starting from 1.



3. Calculate grades of being relevant and retrieved. All matched pairs are elements of two sets of which one is fuzzy. All non-matched detected and hand segmented boundaries are elements of one set. Let  $G$  denote the set of relevant (correct) elements. Let  $A$  denote the ordered set containing retrieved (predicted) boundaries. For each segmentation boundary  $x$  in  $A$  be define a fuzzy membership function  $f(x)$  that describes the degree to which  $x$  has been accurately segmented. There are three different scenarios for calculating membership function  $f(x)$ :

- A hand segmented boundary not matched with any detected boundary is an element of set  $G$ .
- A detected boundary  $x$  not matched with any hand segmented boundary is an element of set  $A$  and has  $f(x) = 0$ . The last detected boundary on the fig. 3 is such a case.
- If the detected boundary  $x$  is inside the hand segmented boundary range the boundary is the element of both sets  $A$  and  $G$ . The other probabilistic factor is boolean and represents membership of a set with hand segmentation boundaries. We use algebraic product of these two probabilistic grades as a  $T$ -norm, to find a membership grade of the intersection. In the situation where  $x$  is inside the hand segmented boundary, range  $f(x) = 1$ .
- Otherwise it is a fuzzy case and  $f(x) = a - b/a$  where  $a$  stands for the half of the length of the phoneme which the boundary was detected (take the phoneme in which the detected boundary is situated) and  $b$  stands for the distance between hand segmented boundary and the detected one (fig. 4). All boundaries on the fig. 3 apart from the last one are examples of this case, which

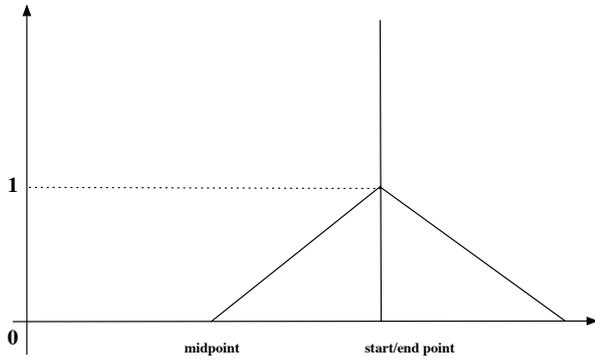


Figure 4: Fuzzy membership

proves how fuzzy logic can be useful in the segmentation evaluation.

4. Fuzzy precision can be calculated as

$$P = \frac{\sum_{x \in A} f(x)}{|G|}. \quad (1)$$

5. Fuzzy recall equals

$$R = \frac{\sum_{x \in A} f(x)}{|A|}. \quad (2)$$

Recall and precision can be used to give a single evaluation grade in many different ways according to which of them is more important. Widely used way is calculating *F*-score (van Rijsbergen, 1979)

$$F = \frac{(\beta^2 + 1) * P * R}{(\beta^2 * P) + R}, \quad (3)$$

where  $\beta$  is a parameter to the *F*-score. Often  $\beta = 1$ , that is, precision and recall are given equal weights. Higher  $\beta$  values would favour recall over precision.

## 5. Comparison with other methods

Evaluation methods are always subjective and there is no way to grade them statistically. This is why it is difficult to compare evaluation methods and judge which one is better. As we cannot prove our method outperforms the others we present an example which might explain why we believe so. There is no standard method but all evaluations are based on insertion and deletion with some tolerances. We compare using such methods with the fuzzy recall and precision for the example presented in fig. 3. Indexes are due to segmentation method (Ziółko et al., 2006). One index unit corresponds to 5.8 ms. The very first and last boundary is not included due to assumption that they are supposed to be perfectly detected. The Table 1 list membership function  $f(x)$  for all boundaries. In lower rows insertions and deletions with all possible tolerances are marked. The symbol X stands for a boundary with a deletion or insertion for a given tolerance, while  $\checkmark$  stands for a boundary accepted as a correct one with a given tolerance. The number of insertions and deletions is given in brackets in the first column.

beg	9	56	89	113	156	196	-
end	10	58	90	114	158	198	-
auto	15	59	97	112	159	195	206
fuzzy recall and precision							
f(x)	0.78	0.93	0.36	0.91	0.95	0.95	0
insertions and deletions without tolerance							
Ins(7)	X	X	X	X	X	X	X
Del(6)	X	X	X	X	X	X	-
with tolerance from 1 (5.8 ms) to 4 (23.2 ms) - same results							
Ins(3)	X	$\checkmark$	X	$\checkmark$	$\checkmark$	$\checkmark$	X
Del(2)	X	$\checkmark$	X	$\checkmark$	$\checkmark$	$\checkmark$	-
with tolerance 5 (29 ms) or 6 (34.8 ms)							
Ins(2)	$\checkmark$	$\checkmark$	X	$\checkmark$	$\checkmark$	$\checkmark$	X
Del(1)	$\checkmark$	$\checkmark$	X	$\checkmark$	$\checkmark$	$\checkmark$	-
with tolerance 7 (40.6 ms) or higher							
Ins(1)	$\checkmark$	$\checkmark$	$\checkmark$	$\checkmark$	$\checkmark$	$\checkmark$	X
Del(0)	$\checkmark$	$\checkmark$	$\checkmark$	$\checkmark$	$\checkmark$	$\checkmark$	-

Table 1: Comparison of fuzzy recall and precision with commonly used methods based on insertions and deletions for an exemplar word.

As we use only a single word, results are the same for many tolerance levels. For a larger corpora it would not take a place. It is clearly visible that counting insertions and deletions is less accurate unless one uses tolerance levels with resolution equals to the resolution of index order. Especially using single tolerance level smooths information about boundary detections. Perfectly accurate detections are graded in the same way as unperfect but fulfilling a tolerance level. Using several tolerance levels improves but is still just a step in high resolution evaluation method as suggested fuzzy recall and precision. Another issue is the length of phonemes. The method based on tolerances gives grade without comparing the tolerance and the given phoneme length. In other words our method is better because the membership function  $f(x)$  is calculated on percentage of the phoneme length the boundary was missed and not on the constant tolerance value. In the presented example, phoneme lengths vary from 11 (64 ms) to 47 (273 ms). The tolerance of i.e. 3 (17 ms) is effectively much higher for the shortest unit than for the longest one. There is no such flaw in our method. The algorithm implemented in C++ is available on <http://WWW-users.cs.york.ac.uk/~bziolko/>. Final grades for a given word are: precision: 0.813901, recall: 0.697629, f-score: 0.751293.

## 6. Conclusions

The precise evaluation method was described. It adapts a standard and very useful recall and precision scheme for applications where evaluation has to consider more details. Speech segmentation is such a field, however, many other types of segmentation are as well. The reason is that the correctness of audio or image segmentation is typically not binary. This is why we found usefulness of fuzzy sets in the task of segmentation evaluation. General rules of applying fuzzy logic into recall and precision were presented as well as exact algorithm of using it for phoneme segmentation evaluation, as an example.

## 7. References

- Cardinal, P., G. Boulianne, and M. Comeau. Segmentation of recordings based on partial transcriptions. *Proceedings of Interspeech 2005*:3345–3348.
- Demuynck, K. and T. Laureys, 2002. A comparison of different approaches to automatic speech segmentation. *Proceedings of the 5th International Conference on Text, Speech and Dialogue*:277–284.
- Grayden, D. B. and M. S. Scordilis, 1994. Phonemic segmentation of fluent speech. *Proceedings of ICASSP*:73–76.
- Holmes, J. N., 2001. *Speech Synthesis and Recognition*. London: Taylor and Francis.
- Kecman, V., 2001. *Learning and Soft Computing*. US: Massachusetts Institute of Technology.
- Ostendorf, M., V. V. Digalakis, and O. A. Kimball, 1996. From HMM's to segment models: A unified view of stochastic modeling for speech recognition. *IEEE Transactions on Speech and Audio Processing*, 4:360–378.
- Subramanya, A., J. Bilmes, and C. P. Chen, 2005. Focused word segmentation for ASR. *Proceedings of Interspeech 2005*:393–396.
- Suh, Y. and Y. Lee, 1996. Phoneme segmentation of continuous speech using multi-layer perceptron. In *Proceedings of ICSLP*.
- Tan, B. T., R. Lang, H. Schroder, A. Spray, and P. Dermody, 1994. Applying wavelet analysis to speech segmentation and classification. *H. H. Szu, editor, Wavelet Applications, volume Proc. SPIE 2242*:750–761.
- van Rijsbergen, C. J., 1979. *Information Retrieval*. London: Butterworths.
- Villing, R., J. Timoney, T. Ward, and J. Costello, 2004. Automatic blind syllable segmentation for continuous speech. *Proceedings of ISSC 2004, Belfast*.
- Young, S., 1996. Large vocabulary continuous speech recognition: a review. *IEEE Signal Processing Magazine*, 13(5):45–57.
- Zheng, C. and Y. Yan, 2004. Fusion based speech segmentation in DARPA SPINE2 task. *Proceedings of ICASSP 2004*:I-885–888.
- Ziółko, B., S. Manandhar, R. C. Wilson, and M. Ziółko, 2006. Wavelet method of speech segmentation. *Proceedings of 14th European Signal Processing Conference EUSIPCO*.