# Fishing for meaningful units in connected speech

Peter Juel Henrichsen[1] and Thomas Ulrich Christiansen[2]

[1] *Center for Computational Modelling of Language (CMOL), Copenhagen Business School, DK-2000 Frederiksberg, Denmark*

[2] *Centre for Applied Hearing Research (CAHR), Technical University of Denmark, DK-2800 Lyngby, Denmark*

In many branches of spoken language analysis including Automatic Speech Recognition (ASR), the set of smallest *meaningful* units of speech is taken to coincide with the set of phones or phonemes. However, fishing for phones is difficult, error-prone, and computationally expensive. We present an experiment, based on machine learning, with an alternative approach. Instead of stipulating a basic set of target units, the determination of the set is considered to be part of the learning task. Given 18 recordings of Danish talkers performing a simple lab task, our algorithm produced a set of acoustically well-defined units sufficient for identifying all the major semantic elements (be they parts of words, single words or several words), relevant to the task. As the sound encoding used was very simple – fundamental frequency (F0), Harmonicity-to-Noise-Ratio (HNR), and Intensity samples only – the computational complexity involved was far lower than for phonemic recognition. Our findings show that it is possible to automatically characterize a linguistic message, without detailed spectral information or presumptions about the target units. Further, fishing for simple meaningful cues and enhancing these selectively would potentially be a more effective way of achieving intelligibility transfer, which is the end goal for speech transducing technologies.

## INTRODUCTION

A speech signal is a succession of acoustic-phonetic events – some abrupt, some short-lived, some characterised by a period of relative stability. Deciding what events should count as *meaningful* is a purpose-driven activity. If the purpose is a correct identification of the lexemes (words) represented in the speech stream, the formal setup is equivalent to that of the mainstream ASR. Current ASR technology is fairly successful when confined to semantically restricted domains, such as medical, scientific, or legal jargon where the vocabulary is well-delimited and where subtle intonational variations can be left out of consideration. When broadening the pragmatic scope to free-style dialogue, the identification of meaningful units becomes immensely intractable, arguably approaching AI completeness. In every-day conversation, the tiniest variations in sound can of course represent huge differences in conversational implicature. Just imagine what changing the intonation of the word yes can do to a question like “will you marry me?”.

In this paper we present a practical experiment in automatic identification of speech events. For reasons of tractability, we settled on data from a description task with

Proceedings of ISAAR 2009: Binaural Processing and Spatial Hearing. 2nd International Symposium on Auditory and Audiological Research. August 2009, Elsinore, Denmark. Edited by J. M. Buchholz, T. Dau, J. Cristensen-Dalsgaard, and T. Poulsen. ISBN: 87-990013-2-2. The Danavox Jubilee Foundation, 2010.

very limited degrees of linguistic liberty. The speech recordings were made in lab sessions with subjects describing the colors, shapes, and spatial relations of the figures in a simple drawing. Our objective, then, was to show that the raw sound data supplemented by a formal representation of the drawing would suffice for an inference engine to correctly identify the color terms and shape terms as they occurred in the speech stream. No pre-segmentation or manual annotation of the sound stream was thus involved. The recordings were analysed in three conceptually basic and computationally tractable acoustic parameters: intensity, HNR (Boersma, 1993), and fundamental frequency (F0).

In the following sections, we first introduce the speech data and the acoustic analysis employed. We then present the inference system responsible for the mapping between sound events and semantic constituents. Finally we broaden the discussion by suggesting a new perspective for the development of speech enhancing devises.

Due to space limitations, we must skip many of the technical details. Most parts of the formal framework will appear in simplified versions; some are presented by examples only.

## THE OVERALL GOAL OF THE EXPERIMENT

Given a plain-language description of a simple illustration, identify and classify the segments of the speech stream that denote the various constituent elements. The processing must be automatic (i.e. unsupervised) and based entirely on acoustic measurement, statistical analysis, and logical inference.

## THE SPEECH MATERIAL

Our starting point was the Danish corpus DanPASS (Grønnum 2009). Corpus DanPASS, short for Danish Phonetically Annotated Spontaneous Speech, is a collection of recordings of Danish subjects performing a number of traditional phonologist's lab tasks such as map guidance, construction reports, and descriptions of simple graphical layouts. The recordings were made in the non-echoic chamber of KUA (University of Copenhagen) using high quality B&K microphones. DanPASS has two subparts, the monologues and the dialogues, the former representing 18 adult male and female native speakers of Danish. We opted for this particular description task for these reasons:

- each recording has a very restricted number of content word types[1],
- content words have specific and consistent denotations,
- content words have multiple occurrences in each recording.

[1] Content word is the linguistic term for semantically 'autonomous' words, mainly nouns, verbs, and adjectives, which often do not depend on other words for their denotation (as opposed to e.g. conjunctions, pronouns, and prepositions).

Transcription sample from DanPASS (describing Fig. 1).

I will start + down at the bottom of the figure with a blue + square + then I will go one step up + and take a green + and there will be a green circle + take one step up again + there is a + purple triangle

Commas designate stress, plus designate pause, equal sign designates hesitation. Notice the relaxed sentence syntax typical of spontaneous speech. The complete transcriptions and background information can be found at www.danpass.dk (permission required).

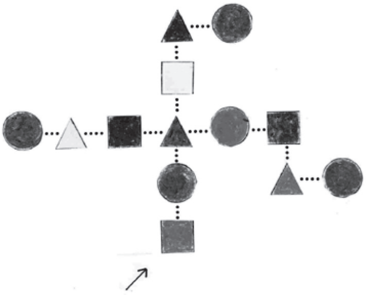

**Fig. 1**: The geometrical layout. Subjects were instructed to give a complete description of the network beginning with the bottom square by the arrow. The illustration is adopted from Grønnum (2009). The original figure is in color. The geometrical objects are blue, green, yellow, red, brown or purple. Some shapes occur more than once in the same color.

## ACOUSTIC PARAMETERS AND NUMERICAL ANALYSIS

### Acoustic parameters

DanPASS was recorded with a sampling rate of 44.1 kHz. From this the three parameters used in this study F0, HNR and intensity were calculated in 5 ms frames.

F0 is an estimate of the frequency of vocal fold vibration, i.e., the inverse pitch period, which in turn corresponds to the position of the maximum of the autocorrelation function of the signal. HNR is the degree of periodicity as designated by the relative height of this maximum given in dB. Details about the method used for calculating F0 and HNR is described in Boersma (1993). Praat calculates intensity in dB re $10^{12}$ watts/$m^2$ based on RMS and an absolute reference level from then recording.

### Data representation

Figure 2 shows the acoustic data representing the first sounds of the Danish unstressed article “den” (*the*), in phonetic terms a stop consonant followed by a full vowel: [dE].

### Some observations

The acoustic features and the corresponding phonetic speech elements are often straightforwardly correlated. Consider some examples.

- Intensity, *t*=1.875 ms: stopped sound correspond to a local Intensity minimum corresponding to a 2nd derivative maximum.
- HNR, *t*=1.930 ms: full vowel correspond to a local maximum for HNR (2nd derivative minimum).
- F0, *t*=1.940 ms: unstressed vowel correspond to a steady F0 drop (1st derivative minimum, 2nd derivative approaching zero).

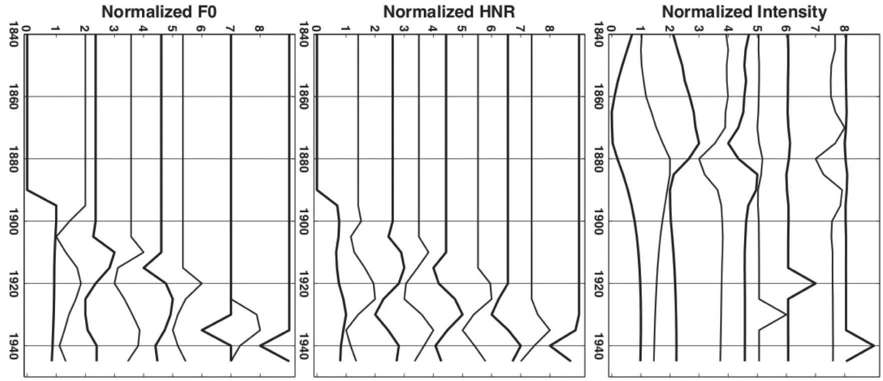

**Fig. 2**: The acoustic parameters F0, HNR, Intensity and derivatives as a function of time. The vertical axis represents the abscissa, which is time in ms starting at t=1840 ms. Each of the horizontal axes represents ordinates with normalized values between 0 and 1 of each of the acoustic parameter derivatives. The functions are offset so the minimum of any given function is aligned with the ordinate value designating the derivative order e.g. the leftmost function represents the 0'th derivative of the normalized F0 i.e. F0, the adjacent function represents the first derivative etc.

## DISCRETIZATION OF THE SPEECH STREAM

Generalizing the observations above, we introduce the concept of *proto-phones*. Informally, a proto-phone is an acoustic pattern visible in the dimensions of F0, HNR, and Intensity (including their derivatives), anchored to a specific time frame. The phonetic class of stopped consonants thus corresponds to the proto-phone formally defined as in Fig. 3.

| | | | | | |
|---|---|---|---|---|---|
| F0: | [f : *undefined* | f' : *undefined* | f'' : *undefined* | f''' : *undefined* | ...] |
| HNR: | [f : *undefined* | f' : *undefined* | f'' : *undefined* | f''' : *undefined* | ...] |
| INT: | [f: *defined* | f' : 0 | f'' : *positive* | f''' : *defined* | ...] |

**Fig. 3**: Proto-phone corresponding to the Danish stopped consonant where f represents the measured value in each parameter while f’, f’’, f’’’, ... represent the 1st, 2nd, 3rd, ... derivatives.

The general scheme is that proto-phones are determined as combinations of values for each of the three parameters including their derivatives up to a certain limit. The permitted values are defined by a value space *VSpace*.

Vspace = {*unspecified*, *defined*, *undefined*, 0, *non*-0, *positive*, *negative*}

The derivational limit selected determines the granularity of the speech segmentation, or in other words the time window in which each speech event is confined. Setting the limit to, say, 8 derivatives, detectable speech events can have durations up to eight time frames before and eight time frames after the focus frame, or 17 frames in total. In our current setup, 17 frames correspond to a time window of 85 ms – a reasonable choice for studying phonemic events like stops, fricatives, and vowels.

The set of possible proto-phones is, of course, much larger than the set of symbols used in traditional phonetic transcriptions. While most combinations are extremely unlikely to ever occur in an actual Danish speech signal, some are expected to occur in other languages. Examples are the tongue clicks of South African languages like Xhosa. The click is characterized by a local intensity maximum corresponding to a proto-phone like the one in Fig. 3 except that the f’’ value for INT *is negative.*

Yet other proto-phone types are found in non-linguistic sound instances only, such as the rattling noise of a paper sheet being manipulated by a test person, a beep from a watch, a chair being moved, and so forth. Signal decoding at the level of the proto-phones may thus provide a basis for linguistically informed sound transduction in speech enhancing devises; more on this in the final section.

Each sound recording was segmented into proto-phone events using statistical best-fit methods. The algorithm is presented here only very briefly. For each time frame, two factors are determined, (i) the proto-phone configuration *p* that provides the closest match to the acoustic measurements, and (ii) a temporal likelihood factor *t* based on expected proto-phone duration (a distribution function defined independently of the particular sound signal). Finally, the anchoring of proto-phone events to time-frames is performed by maximizing *p·t*. Further details are to be published elsewhere.

## LINGUISTIC CLASSIFICATION

The speech streams, now discretized at the level of the proto-phones, were then further segmented into potentially meaningful units using the Siblings-and-Cousins algorithm of Henrichsen (2004). Originally, the S-and-C algorithm was suggested as a way of organizing the lexical items (words) occurring in an unannotated text corpus, in clusters based on their distributional similarity. A main ingredient in the S-and-C framework is the *proximity* measure comparing the similarity of two types based on the distribution of their adjacent tokens (left and right) in the corpus. Modifying the algorithm slightly to accommodate the current data type, the proximity of two proto-phone *n*-grams[2] *X* and *Y* is given by:

$$Prox(X,Y,S)=\frac{\sum_{z\in Voc} C_z\cdot\left(1-\frac{(L_1-L_2)}{L_1+L_2}\right)}{C_x}\cdot\frac{\sum_{z'\in Voc} C_{z'}\cdot\left(1-\frac{(R_1-R_2)}{R_1+R_2}\right)}{C_x} \qquad \text{(Eq. 1)}$$

where *S* is the sound signal segmented into proto-phones, *Voc* is the set of all *n*-proto-phone types in *S,* $L_1$ is the number of occurrences in *S* of the substring [*z X*], $L_2$ of [*z Y*], $R_1$ of [*X z'*], and $R_2$ of [*Y z'*], $C_x$, $C_z$ and $C_{z'}$ is the number of occurrences in *S* of types *X, z*, and *z'*, respectively. Proximity values range between 0 and 1 (for valid input). Kindred *n*-grams score high, unrelated *n*-grams score low.

Proximity($N_{\text{trekant}}$ , $N_{\text{cirkel}}$ , *S13*) = 0.646 (*HIGH PROXIMITY*)
Proximity($N_{\text{trekant}}$ , $N_{\text{gul}}$ , *S13*) = 0.284 (*MEDIUM PROXIMITY*)
Proximity($N_{\text{trekant}}$ , $N_{\text{over}}$ , *S13*) = 0.056 (*LOW PROXIMITY*)

In these examples, $N_{\text{trekant}}$ is short for the proto-phone *n*-gram representing the type ‘trekant’ (*triangle*) as pronounced by the speaker represented as *S13*. Similarly for $N_{\text{cirkel}}$ (*circle*), $N_{\text{gul}}$ (*yellow*), and $N_{\text{over}}$ (*over*). As the example suggests, word types belonging to the same semantic category, e.g. color terms, shape terms, or direction terms, typically score high while unrelated pairs like ‘trekant’ + ‘over’ score low. Using the proximity scores for all pairs of proto-phone *n*-grams, semantic equivalence classes were derived with simple statistical methods.

| | | |
|---|---|---|
| `1.000000` | `t_r_{_k_a_n_?_d` | "trekant" (*triangle*) |
| `0.837132` | `f_i_R_k_a_n_?_d` | "firkant" (*square*) |
| `0.727861` | `l_e_l_a_t_r_{_k_a_n_?_d` | "lilla trekant" (*purple triangle*) |
| `0.646050` | `s_i_R_g_l` | "cirkel" (*circle*) |
| `0.629778` | `t_r_{_k_a_n_?_t` | "trekant" (*triangle*) |

In this sample from the equivalence class of *X*=’trekant’ (*triangle*), proximity values appear on the left. Only the four top-ranked (i.e. most cognate) *n*-grams are shown, and the proto-phones have been replaced by their nearest phonetic equivalents (SAMPA-style, www.phon.ucl.ac.uk/home/sampa/danish.htm) for perspicuity. Notice that two distinct phonetic realizations of “trekant” are present. Otherwise, the derived class faithfully represents the shape terms.

[2] In linguistic terminology, an *n*-gram is a number of adjacent segments in a sequence of tokens (e.g. words, phonemes, or proto-phones). Special cases are the bigram (*n*=2), trigram (*n*=3), and soforth. A single token is its own monogram (*n*=1).

Observe, however, that the *n*-gram of ‘lilla trekant’ (*purple triangle*) has been included among the shape terms, perhaps unexpectedly. Since the drawing (Fig. 1) has only a single purple figure, *purple* is in this particular setting void of information, hence cannot function as a distinguishing feature in the classifying regime. This is an illustrative example of speech unit identification as a purpose-driven activity.

## THE INFERENCE ENGINE

An inference engine performs the mapping of speech units on semantic constituent properties (color and shape terms), based on a formal representation of the geometrical network. The algorithm is implemented in the programming language PROLOG[3]. In this language, propositional knowledge is particularly easy to formalize and to reason about.

```
property(color,blue,[e1,e10,e12]).
property(color,green,[e2,e6,e9,e13]).
property(color,red,[e4,e8]).
property(color,yellow,[e5,e7]).
property(color,purple,[e3]).
property(color,brown,[e11]).
property(shape,square,[e1,e4,e7,e11]).
property(shape,circle,[e2,e6,e9,e10,e13]).
property(shape,triangle,[e3,e5,e8,e12]).
```

**Fig. 4**: Formal representation of the geometrical layout (Fig. 1), expressed in PROLOG clauses.

The symbols e1, e2, ... , e13 in Fig. 4 are semantic constants referring to the thirteen figures of the geometrical layout (Fig. 1). Each symbol belongs to one color set and to one shape set. e1 thus denotes the blue square (the starting point of the descriptions) while the purple triangle is e3.

The test subjects were instructed to name all figures. Though no clues were given as to the order of the enumeration, the subjects seemed to follow an adjacency policy proceeding from each figure to a neighboring one whenever possible. Figures e1, e2, and e3 were thus always named first, and in that order. The central purple triangle marks a point of choice, leaving six optimal paths through the possibility space. We therefore expected the subjects to select their naming strategy among A-F.

```
path('A',[e1,e2,e3,e4,e5,e6,e7,e8,e9,e10,e11,e12,e13]).  % up - left - up - rigth
path('B',[e1,e2,e3,e4,e5,e6,e10,e11,e12,e13,e7,e8,e9]).  % up - left - right - up
path('C',[e1,e2,e3,e7,e8,e9,e4,e5,e6,e10,e11,e12,e13]).  % up - up - left - right
path('D',[e1,e2,e3,e7,e8,e9,e10,e11,e12,e13,e4,e5,e6]).  % up - up - right - left
path('E',[e1,e2,e3,e10,e11,e12,e13,e4,e5,e6,e7,e8,e9]).  % up - right - left - up
path('F',[e1,e2,e3,e10,e11,e12,e13,e7,e8,e9,e4,e5,e6]).  % up - right - up - left
```

[3] Prolog is a programming language based on logical inference rather than assertive commands (e.g. Bratko, 2000). Prolog is often used for AI and for applications for deductive reasoning.

In consequence, the objective for the inference program was to select correctly among the paths A-F for each subject based on the speech unit classes imported from the S-and-C application. To our surprise and, initially, disappointment the program found only 14 paths. On closer inspection of the sound recordings, however, we discovered that two of the subjects had actually chosen alternative (and less straight) paths[4] not in A-F while the remaining two made a naming error each. When compensating for these irregularities, the inference engine hit 100% precision.

## CONCLUDING REMARKS

So, what has been obtained by this toyish experiment? Of course, fishing for meaningful units in discourse like the circle-and-triangle monologues is far less complex than searching for pragmatic cues in a realistic conversational scene. This being said, our experiment did show that complex phonetic units could be identified, categorized for semantic denotation, and even used for a 'practical' purpose (path detection) in a radically data-driven setup not presuming any knowledge about the phonetics, vocabulary, or semantic categories of the Danish language as such. In contrast, semantically active parts of the speech sound were detected based on low-level acoustic cues only, yet providing enough data for an inference machine to perform a high-level semantic decision task.

The human brain is the ultimate inference machine. Inspired by our test results, we speculate that the development of speech transmitting devises could benefit from a change in strategy. Instead of amplifying uncritically the sound signal as such, fishing for simple meaningful cues and enhancing these selectively would provide better data for the receiving mind in its reconstruction of the intended message. The overall goal for speech transducing technologies like hearing aids and telecommunication would then be redefined from signal transfer to intelligibility transfer.

## REFERENCES

Boersma, P. (**1993**). "Accurate short-term analysis of the fundamental frequency and the harmonics-to-noise ratio of a sampled sound," IFA Proceedings **17**, 97-110.

Boersma, P. (**2001**). "Praat, a system for doing phonetics by computer", Glot. International **5**:9/10, 341-345.

Bratko, I. (**2000**). *Prolog Programming for Artificial Intelligence*, Third Edition (Addison-Wesley E).

Grønnum, N. (**2009**). "A Danish phonetically annotated spontaneous speech corpus (DanPASS)", Speech Communication **51**, 594-603.

Henrichsen, P. J. (**2004**). "Siblings and Cousins; Statistical Methods for Spoken Language Analysis", Acta Linguistica Hafniensia **36**, 7-33.

[4] path('G',[e1,e2,e3,e7,e8,e9,e10,e11,e4,e5,e6,e12,e13]) + path('H',[e1,e2,e3,e7,e8,e10,e11,e4,e5,e6,e9,e12,e13]).