

Go to the [first](#), [previous](#), [next](#), [last](#) section, [table of contents](#).

32 Feature functions

This chapter contains a list of a basic feature functions available for stream items in utterances. See section [14.6 Features](#). These are the basic features, which can be combined with relative features (such as `n.` for next, and relations to follow links). Some of these features are implemented as short C++ functions (e.g. `asyl_in`) while others are simple features on an item (e.g. `pos`). Note that functional feature take precedence over simple features, so accessing and feature called "X" will always use the function called "X" even if a the simple feature call "X" exists on the item.

Unlike previous versions there are no features that are builtin on all items except `addr` (reintroduced in 1.3.1) which returns a unique string for that item (its the hex address on teh item within the machine). Features may be defined through Scheme too, these all have the prefix `lisp_`.

The feature functions are listed in the form *Relation.name* where *Relation* is the name of the stream that the function is appropriate to and *name* is its name. Note that you will not require the *Relation* part of the name if the stream item you are applying the function to is of that type.

`ANY.addr`

Returned by popular demand, returns the address of given item that is guaranteed unique for this session.

`ANY.lisp_*`

Apply Lisp function named after `lisp_`. The function is called with an stream item. It must return an atomic value. This method may be inefficient and is primarily desgined to allow quick prototyping of new feature functions.

`Intonation.lisp_last_tilt_accent`

Returns the most recent tilt accent.

`Intonation.lisp_last_tilt_boundary`

Returns the most recent tilt boundary.

`Intonation.lisp_next_tilt_accent`

Returns the next tilt accent.

`Intonation.lisp_next_tilt_boundary`

Returns the next tilt boundary.

`Intonation.peak_anchor_segment_type ie`

Determines whether the segment anchor for a peak is the first consonant of a syl - C0 -, the vowel of a syl - V0 -, or segments after that - C1->X,V1->X. If the segment is in a following syl, the return value will be preceded by a 1 - e.g. 1V1

`Segment.diphone_phone_name`

This is produced by the diphone module to contain the desired phone name for the desired diphone.

This adds things like `_` if part of a consonant or `$` to denote syllable boundaries. These are generated on a per voice basis by function(s) specified by `diphone_module_hooks`. Identification of dark ll's etc. may also be included. Note this is not necessarily the name of the diphone selected as if it is not found some of these characters will be removed and fall back values will be used.

`Segment.lisp_pos_in_syl seg`

Finds the position in a syllable of a segment - returns a number.

`Segment.ph_*`

Access phoneset features for a segment. This definition covers multiple feature functions where `ph_` may be extended with any features that are defined in the phoneset (e.g. `vc`, `vlng`, `cplace` etc.).

`Segment.pos_in_syl`

The position of this segment in the syllable it is related to. The index counts from 0. If this segment is not related to a syllable this returns 0.

`Segment.seg_coda_fric`

Returns 1 if coda of the syllable this segment is in contains a fricative. 0 otherwise.

`Segment.seg_onset_stop`

Returns 1 if onset of the syllable this segment is in contains a stop. 0 otherwise.

`Segment.seg_onsetcoda`

Returns onset if this segment is before the vowel in the syllable it is contained within. Returns coda if it is the vowel or after. If the segment is not in a syllable it returns onset.

`Segment.seg_pitch`

Pitch at the middle of this segment.

`Segment.segment_duration`

The duration of the given stream item calculated as the end of this item minus the end of the previous item in the Segment relation.

`Segment.segment_end`

The end time of the given segment.

`Segment.segment_mid`

The middle time of the given segment.

`Segment.segment_start`

The start time of the given segment.

`Segment.syl_final`

Returns 1 if this segment is the last segment in the syllable it is related to, or if it is not related to any syllable.

`Segment.syl_initial`

Returns 1 if this segment is the first segment in the syllable it is related to, or if it is not related to any syllable.

`Syllable.accented`

Returns 1 if syllable is accented, 0 otherwise. A syllable is accented if there is at least one IntEvent related to it.

`Syllable.asyl_in`

Returns number of accented syllables since last phrase break, not including this one. Accentedness is as defined by the `syl_accented` feature.

`Syllable.asyl_out`

Returns number of accented syllables to the next phrase break, not including this one. Accentedness is as defined by the `syl_accented` feature.

`Syllable.last_accent`

Returns the number of syllables since last accented syllable.

`Syllable.lisp_last_stress`

Number of syllables from previous stressed syllable. 0 if this syllable is stressed. It is effectively assumed that the syllable before the first syllable is stressed.

`Syllable.lisp_next_stress`

Number of syllables to next stressed syllable. 0 if this syllable is stressed. It is effectively assumed the syllable after the last syllable is stressed.

`Syllable.lisp_tilt_accent`

Returns "a" if there is a tilt accent related to this syllable, 0 otherwise.

`Syllable.lisp_tilt_accented`

Returns 1 if there is a tilt accent related to this syllable, 0 otherwise.

`Syllable.lisp_tilt_boundaried`

Returns 1 if there is a tilt boundary related to this syllable, 0 otherwise.

`Syllable.lisp_tilt_boundary`

Returns boundary label if there is a tilt boundary related to this syllable, 0 otherwise.

`Syllable.lisp_time_to_next_vowel syl`

The time from `vowel_start` to next `vowel_start`

`Syllable.next_accent`

Returns the number of syllables to the next accented syllable.

`Syllable.old_syl_break`

Like `syl_break` but 2 and 3 are promoted to 4 (to be compatible with some older models).

`Syllable.pos_in_word`

The position of this syllable in the word it is related to. The index counts from 0. If this syllable is not related to a word then 0 is returned.

`Syllable.position_type`

The type of syllable with respect to the word it is related to. This may be any of: single for single syllable words, initial for word initial syllables in a poly-syllabic word, final for word final syllables in poly-syllabic words, and mid for syllables within poly-syllabic words.

`Syllable.ssylin`

Returns number of stressed syllables since last phrase break, not including this one.

`Syllable.ssylin_out`

Returns number of stressed syllables to next phrase break, not including this one.

`Syllable.stress`

The lexical stress of the syllable as specified from the lexicon entry corresponding to the word related to this syllable.

`Syllable.sub_phrases`

Returns the number of non-major phrase breaks since last major phrase break. Major phrase breaks are 4, as returned by `syl_break`, minor phrase breaks are 2 and 3.

`Syllable.syl_accent`

Returns the name of the accent related to the syllable. NONE is returned if there are no accents, and multi is returned if there is more than one.

`Syllable.syl_break`

The break level after this syllable. Word internal is syllables return 0, non phrase final words return 1. Final syllables in phrase final words return the name of the phrase they are related to. Note the occasional "-" that may appear of phrase names is removed so that this feature function returns a number in the range 0,1,2,3,4.

`Syllable.syl_coda_type`

Return the van Santen and Hirschberg classification. -V for unvoiced, +V-S for voiced but no sonorants, and +S for sonorants.

`Syllable.syl_codasize`

Returns the number of segments after the vowel in this syllable. If there is no vowel in the syllable this will return the total number of segments in the syllable.

`Syllable.syl_endpitch`

Pitch at the end of this syllable.

`Syllable.syl_in`

Returns number of syllables since last phrase break. This is 0 if this syllable is phrase initial.

`Syllable.syl_midpitch`

Pitch at the mid vowel of this syllable.

`Syllable.syl_numphones`

Returns number of phones in syllable.

`Syllable.syl_onset_type`

Return the van Santen and Hirschberg classification. -V for unvoiced, +V-S for voiced but no sonorants, and +S for sonorants.

`Syllable.syl_onsetsize`

Returns the number of segments before the vowel in this syllable. If there is no vowel in the syllable this will return the total number of segments in the syllable.

`Syllable.syl_out`

Returns number of syllables to next phrase break. This is 0 if this syllable is phrase final.

`Syllable.syl_pc_unvox`

Percentage of total duration of unvoiced segments from start of syllable. (i.e. percentage to start of first voiced segment)

`Syllable.syl_startpitch`

Pitch at the start of this syllable.

`Syllable.syl_vowel`

Returns the name of the vowel within this syllable. Note this is not the general form you probably want. You can't refer to `ph_*` features of this. Returns "novowel" if no vowel can be found.

`Syllable.syl_vowel_start`

Start position of vowel in syllable. If there is no vowel the start position of the syllable is returned.

`Syllable.syllable_duration`

The duration of the given stream item calculated as the end of last daughter minus the end of previous item in the Segment relation of the first daughter.

`Syllable.word_end`

The end time of the given syllable.

`Syllable.syllable_start`

The start time of the given syllable.

`Syllable.tobi_accent`

Returns the ToBI accent related to syllable. ToBI accents are those which contain a *. NONE is returned if there are none. If there is more than one ToBI accent related to this syllable the first one is returned.

`Syllable.tobi_endtone`

Returns the ToBI endtone related to syllable. ToBI end tones are those IntEvent labels which contain a % or a - (i.e. end tones or phrase accents). NONE is returned if there are none. If there is more than one ToBI end tone related to this syllable the first one is returned.

`Syllable.lisp_get_onset_length`

`Syllable.lisp_get_rhyme_length`
Length from start of syllable to start of vowel.

`SylStructure.lisp_length_to_last_seg`
Length from start of the vowel to end of syllable.

`SylStructure.lisp_num_postvocalic_c`
Length from start of the vowel to start of last segment of syllable.

`SylStructure.lisp_num_postvocalic_c`
Finds the number of postvocalic consonants in a syllable.

`SylStructure.sonority_scale_coda syl`
Returns value on sonority scale (1 -6, where 6 is most sonorous) for the coda of a syllable, based on least sonorant portion.

`SylStructure.sonority_scale_onset syl`
Returns value on sonority scale (1 -6, where 6 is most sonorous) for the onset of a syllable, based on least sonorant portion.

`SylStructure.lisp_syl_numphones syl`
Finds the number segments in a syllable.

`SylStructure.vowel_frontness syl`
Classifies vowels as front, back or mid

`SylStructure.lisp_vowel_height syl`
Classifies vowels as high, low or mid

`SylStructure.vowel_length syl`
Returns the df.length feature of a syllable's vowel

`Token.prepunctuation`
Preceding punctuation symbol found before token in original string/file.

`Token.punc`
Succeeding punctuation symbol found after token in original string/file.

`Token.whitespace`
Whitespace found before token in original string/file.

`Word.blevel`
A crude translation of phrase break into ToBI like phrase level. Values may be 0,1,2,3,4.

`Word.cap`
Returns 1 if this word starts with a capital letter, 0 otherwise.

`Word.content_words_in`
Number of content words from start this phrase.

`Word.content_words_out`
Number of content words to end of this phrase.

`Word.contentp`
Returns 1 if this word is a content word as defined by gpos, 0 otherwise.

`Word.gpos`
Returns a guess at the part of speech of this word. The lisp a-list guess_pos is used to load up this word. If no part of speech is found in there "content" is returned. This allows a quick efficient method for part of speech tagging into closed class and content words.

`Word.n_content`
Next content word. Note this doesn't use the standard n. notation as it may have to search a number of words forward before finding a non-function word. Uses gpos to define content/function word distinction. This also works for Tokens.

`Word.nn_content`
Next next content word. Note this doesn't use the standard n.n. notation as it may have to search a number of words forward before finding the second non-function word. Uses gpos to define content/function word distinction. This also works for Tokens.

`Word.num_break`
1 if this is the last word in a numeric token and it is followed by a numeric token.

`Word.p_content`
Previous content word. Note this doesn't use the standard p. notation as it may have to search a number of words backward before finding the first non-function word. Uses gpos to define content/function word distinction. This also works for Tokens.

`Word.pbreak`
Result from statistical phrasing module, may be B or NB denoting phrase break or non-phrase break after the word.

`Word.pbreak_score`
Log likelihood score from statistical phrasing module, for pbreak value.

`Word.pos`
Part of speech tag value returned by the POS tagger module.

Word.pos_in_phrase

The position of this word in the phrase this word is in.

Word.pos_score

Part of speech tag log likelihood from Viterbi search.

Word.pp_content

Previous previous content word. Note this doesn't use the standard p.p. notation as it may have to search a number of words backward before finding the first non-function word. Uses gpos to define content/function word distinction. This also works for Tokens.

Word.word_break

The break level after this word. Non-phrase final words return 1 Phrase final words return the name of the phrase they are in.

Word.word_duration

The duration of the given stream item. This is defined as the end of last segment in the last syllable (via the SylStructure relation) minus the segment immediate preceeding the first segment in the first syllable.

Word.word_end

The end time of the given word.

Word.word_numsyls

Returns number of syllables in a word.

Word.word_start

The start time of the given word.

Word.words_out

Number of words to end of this phrase.

Go to the [first](#), [previous](#), [next](#), [last](#) section, [table of contents](#).