

## Hackatorial

Feature Name	Explanation	Example
surface	the word as occurring in the text	Hunde as feature for the word <i>Hunde</i>
surface_backwards	the word spelled backwards	dnuH as feature for <i>Hund</i>
surface_every_second	every second letter of the word	Hn for <i>Hund</i>
word_length	length	4 for <i>Hund</i>
pos	part-of-speech-tag	NN for <i>Hund</i>
pos_first_character	first character of the word's part-of-speech-tag	N for <i>Hund</i> (with the PoS-tag NN)
lemma	the word's canonical/dictionary form	Hund for <i>Hunde</i>
lemma_backwards	lemma spelled backwards	neful for <i>lief</i>
lemma_every_second	lemma's every second letter	lue for <i>lief</i> (with lemma laufen)
segment_id	ID of the segment the word occurs in	===B for a word occurring in the segment named B
sent_id	# of the sentence the word occurs in	10 for a word in the 10th sentence
word_id	position of a word in the sentence	3 for <i>lief</i> in <i>Der Hund lief.</i>
next_word	word immediately after the current word in the same sentence ('None' if it is the last word in the sentence)*	Hund for <i>Der</i> in <i>Der Hund lief.</i> ('None' for <i>lief</i> )
next_verb_in_sent	first verb after the current word in the sentence*	<i>lief</i> for <i>Der</i> (and <i>Hund</i> ) in <i>Der Hund lief und bellte.</i>
next_verb_in_sent_distance	word distance to the next verb within a sentence*	2 for <i>Der</i> in <i>Der Hund lief.</i>
previous_verb_in_sent	verb before the word's occurrence within the same sentence*	<i>lief</i> for <i>und</i> in <i>Der Hund lief und bellte.</i>
previous_verb_in_sent_distance	word distance to the previous verb within the same sentence*	2 for <i>bellte</i> in <i>Der Hund lief und bellte.</i>
next_complete_verb_in_sent	next verb w/ PoS-tag VVFIN within the same sentence*	<i>sprang</i> for <i>Um</i> in <i>Um zu fliehen, sprang er.</i>
next_complete_verb_in_sent_distance	distance to the next verb (PoS-tag VVFIN) within the same sentence*	3 for <i>Um</i> in <i>Um zu fliehen, sprang er.</i>
previous_complete_verb_in_sent	the previous verb (PoS-tag VVFIN) within the same sentence*	<i>sprang</i> for <i>er</i> in <i>Um zu fliehen, sprang er.</i>
previous_complete_verb_in_sent_distance	word distance to the previous verb (PoS-tag VVFIN) within the same sentence*	1 for <i>er</i> in <i>Um zu fliehen, sprang er.</i>
previous_word	the word immediately before the target word (within the same sentence*)	<i>Der</i> for <i>Hund</i> in <i>Der Hund sprang.</i>
next_lemma	lemma of the following word (within the same sentence*)	<i>springen</i> for <i>Hund</i> in <i>Der Hund sprang.</i>
previous_lemma	lemma of the previous word (within the same sentence*)	<i>springen</i> for <i>und</i> in <i>Der Hund sprang und bellte.</i>

\*If the intended feature can't be extracted within the boundaries of the target sentence processed, 'None' will be set as the target word's feature instead.

## Hackatorial

Feature Name	Explanation	Example
next_pos	PoS-tag of the following word (within the same sentence*)	VVFIN for <i>Hund</i> in <i>Der Hund lief.</i>
previous_pos	PoS-tag of the previous word (within the same sentence*)	ART for <i>Hund</i> in <i>Der Hund lief.</i>
find_word_in_sentence74	'True' if the word 'ist' occurs in the 74th sentence, 'False' otherwise	
contains_dot	'True' if the word contains a dot, 'False' otherwise	'True' for the word <i>bw.</i>
contains_dash	'True' if the word contains a dash, 'False' otherwise	'True' for <i>CSU-Fraktion</i>
only_vowels	only the vowels of the word	aue for <i>laufen</i>
only_non_vowels	only non-vowels of a word (including consonants, non-literal-characters)	Hnd for the word <i>Hund</i>
sent_length	length of the sentence containing the word	5 for each word in <i>Der Hund lief und sprang.</i>
more_than_5_chars	'True' if the word consists of more than five characters, 'False' otherwise	'False' for <i>weil</i>
word_is_ascii	'True' if the word only consists of ASCII-characters (characters on an American keyboard), 'False' otherwise;	'False' for €
capitalized	'True' if the word starts with a capital letter, 'False' otherwise	'True' for <i>Hund</i>
previous_word_capitalized	'True' if the previous word (within the same sentence*) starts with a capital letter, 'False' otherwise	'True' for <i>Hund</i> in <i>Der Hund lief.</i>
next_word_capitalized	'True' if the next word (within the same sentence*) starts with a capital letter, 'False' otherwise	'False' for <i>Hund</i> in <i>Der Hund lief.</i>
fifth_word	the fifth word in the corpus	
all_upper_case	'True' if the word only consists of capitalized letters, 'False' otherwise;	'True' for <i>HUND</i>
previous_word_all_upper_case	'True' if the previous word (within the same sentence*) only consists of capitalized letters, 'False' otherwise	'False' for <i>HUND</i> in <i>Ein HUND, wer ...</i>
next_word_all_upper_case	'True' if the next word (within the same sentence*) only consists of capitalized letters, 'False' otherwise	'True' for <i>Ein</i> in <i>Ein HUND, wer ...</i>
suffix	'True' if the word is suffixed by one of the suffixes -ung/-ling/-heit/-schaft/-keit, 'False' otherwise	'True' for <i>Gesundheit</i>
suffix_2	the last two characters of a word	nd for <i>Hund</i>
suffix_3	the last three characters of a word	und for <i>Hund</i>
all_digits	'True' if the word only consists of numbers, 'False' otherwise	'True' for <i>2018</i>
find_Mensch	the word ID of the word if it is 'Mensch', 'None' otherwise	2 for <i>Mensch</i> in <i>Der Mensch rief nach seinem Hund.</i>