standoffconverter

Release 0.5

Oct 14, 2021

Contents

CHAPTER 1

API

class standoffconverter.**Standoff**(*tei_tree*, *namespaces={}*)

Contains a reference to the etree.Element object and the corresponding ContextItem object to link the two representations.

table()

Table as a flattened TEI tree and additional character-position information. The data of the table actually resides at

```
>>> table.df
    position row_type      el  depth  text
0          0     open    text    0.0  None
1          0     open    body    1.0  None
2          0     open       p    2.0  None
3          0     text    None    NaN     1
4          1     text    None    NaN
5          2     text    None    NaN     2
6          3     text    None    NaN
7          4     text    None    NaN     3
8          5    close       p    2.0  None
9          5    close    body    1.0  None
10         5    close    text    0.0  None
```

where the column *position* refers to the character position and *el* is a pointer to the actual etree.Element.

tree()

tree of the TEI XML.

plain()

Plain text string of all text inside the <text> element of the TEI XML.

standoffs()

List of standoff elements of the <text> element fo the TEI XML. Items are traversed in depth-first preorder.

json()

JSON string of standoff elements of the <text> element fo the TEI XML. Items are traversed in depth-first preorder.

collapsed_table()
Table with text and context of the <text> element of the tei tree. All leaf/tail text with the same context is joined.

get_parents(*begin*, *end*, *depth=None*)
Get all parent context.

arguments: begin (int)– beginning character position within the XML end (int)– ending character position within the XML depth (int)– depth of current element

Returns parents (list) – list of parent elements ordered by depth (closest is last).

get_children(*begin*, *end*, *depth*)
Get all children context.

arguments: begin (int)– beginning character position within the XML end (int)– ending character position within the XML depth (int)– depth of current element

Returns children (list) – list of children elements ordered by depth (closest is first).

add_inline(*begin*, *end*, *tag*, *depth=None*, *attrib=None*, *insert_index_at_pos=0*)
Add a standoff element to the structure. The standoff element will be added to the caches and to the etree.

arguments: begin (int)– beginning character position within the XML end (int)– ending character position within the XML tag (str)– tag name, for example 'text' for <text>. depth (int)– depth where to add the element. If None, it will be added deepest attrib (dict)– dictionary of items that go into the attrib of etree.Element. Ultimately, attributes within tags. for example {"resp":"machine"} will result in <SOMETAG resp="machine">.

remove_inline(*del_el*)
Remove a standoff element from the structure. The standoff element will be removed from the caches and from the etree.

arguments: del_el (etree.Element)– the element that should be removed

add_span(*begin*, *end*, *tag*, *depth*, *attrib*, *id_=None*)
Add a span element to the structure. arguments: begin (int)– beginning character position within the XML end (int)– ending character position within the XML tag (str)– tag name, for example 'text' for <text>. depth (int)– depth where to add the element. If None, it will be added deepest

__init__(*tei_tree*, *namespaces={}*)
Create a Converter from a tree element instance.

Parameters **tei_tree** (*etree.Element*) – the etree.Element instance.

Returns The created Standoff instance.

Return type (*Standoff*)

class standoffconverter.**View**(*so*)
Prepare the plain text of a Standoff table for processing with NLP libraries without losing the information of where the characters came from within the Standoff table. Typical use cases are removal of <notes> or insertion of newlines for encoded newlines (<*lb*>).

get_plain()
Plain text of the current status of the view. The plain text output by this function is meant to be inserted into an NLP pipeline. The results of the NLP pipeline will be made on the character level of this sequence. In order to create an annotation within the original TEI, the positions within the TEI that corresponds to the character positions in this plain text sequence can be looked up like this: *view.get_table_pos(plain_text_pos)*.

Returns plain (str)– the plain text str with all modifications applied.

get_table_pos(*plain_text_index*)
the position value within Standoff Table for a given character position. This position can differ from the one in the plain text due to added or removed characters in the plain text (such as multiple whitespace removal or addition of whitespaces at encoded whitespace positions (<*lb/*>)).

arguments: plain_text_index (int)– character position in the plain text output of the view

Returns table_position (int).

get_table_index(*plain_text_index*)
the table_index value for a given character position.

arguments: plain_text_index (int)– character position in the plain text output of the view

Returns table index (int).

exclude_outside(*tag*)
exclude all text outside the tag.

arguments: tag – for example *'note'* or *"{http://www.tei-c.org/ns/1.0}abbr"*

Returns self (standoffconverter.View) for chainability.

exclude_inside(*tag*)
exclude all text within the tag.

arguments: tag – for example *'note'* or *"{http://www.tei-c.org/ns/1.0}abbr"*

Returns self (standoffconverter.View) for chainability.

include_inside(*tag*)
include all text within the tag. It will basically reset all modifications inside the given tag. This means that for example, altered characters or shrunken whitespaces will also be reset. It does not affect any characters outside the given tags (for example, it does not exclude anything outside explicitly). Therefore, it can combined nicely with *exclude_outside*, for example view.exclude_outside("a").include_iside_("b") which will exclude everything except what is inside '<a>'s and '<b>'s.

arguments: tag – for example *'note'* or *"{http://www.tei-c.org/ns/1.0}abbr"*

Returns self (standoffconverter.View) for chainability.

insert_tag_text(*tag*, *text*)
insert a custom character to the plain text for all occurrences of the tag.

arguments: tag – the el tag that should be replaced, for example *"{http://www.tei-c.org/ns/1.0}lb"* text – the text that the el should be replaced with.

Returns self (standoffconverter.View) for chainability.

shrink_whitespace(*shrink_to=' '*, *custom_whitespaces=None*)
Reduce consecutive white spaces to a single white space.

arguments: shrink_to (str)– the character that multiple whitespaces are replaced with (shrunken to). custom_whitespaces (list)– alternative list of characters that are considered as white spaces.

Returns self (standoffconverter.View) for chainability.

remove_comments()
Remove comments (something like "<!– . . . –>") from plain text view.

Returns self (standoffconverter.View) for chainability.

__init__(*so*)
Initialize self. See help(type(self)) for accurate signature.

- demo at https://so.davidlassner.com

- github at https://github.com/standoff-nlp/standoffconverter

CHAPTER 2

Indices and tables

- genindex
- modindex
- search

Index

Symbols

A

C

E

G

I

J

P

R

S

T

V