Package 'finnsurveytext'

March 6, 2025

Type Package

Title Analyse Open-Ended Survey Responses in Finnish

Version 2.1.1

Description Annotates Finnish textual survey responses into CoNLL-U format using Finnish treebanks from <https://universaldependencies.org/format.html> using UDPipe as described in Straka and Straková (2017) <doi:10.18653/v1/K17-3009>. Formatted data is then analysed using single or comparison n-gram plots, wordclouds, summary tables and Concept Network plots. The Concept Network plots use the TextRank algorithm as outlined in Mihalcea, Rada & Tarau, Paul (2004) <https://aclanthology.org/W04-3252/>.

License MIT + file LICENSE

Depends R (>= 2.10)

Imports data.table, dplyr, ggplot2, ggpubr, ggraph, igraph, magrittr, purrr, RColorBrewer, stopwords, stringr, textrank, tibble, tidyr, udpipe, wordcloud

Suggests DT, htmlwidgets, knitr, rmarkdown, shiny, shinyBS, shinydashboard, shinyjs, survey

VignetteBuilder knitr

Encoding UTF-8

LazyData true

RoxygenNote 7.3.2

URL https://dariah-fi-survey-concept-network.github.io/finnsurveytext/, https://github.com/DARIAH-FI-Survey-Concept-Network/finnsurveytext

BugReports https://github.com/DARIAH-FI-Survey-Concept-Network/finnsurveytext/issues

NeedsCompilation no

Author Adeline Clarke [cre, aut], Krista Lagus [aut], Katja Laine [aut], Maria Litova [aut], Matti Nelimarkka [aut],

Joni Oksanen [aut],
Jaakko Peltonen [aut],
Tuukka Oikarinen [aut],
Jani-Matti Tirkkonen [aut],
Ida Toivanen [aut],
Maria Valaste [aut],
Shannon Emilia Carson [ctb],
Sirpa Lappalainen [ctb],
Tuukka Puonti [ctb],
Kimmo Vehkalahti [ctb],
DARIAH-FI [cph, fnd]

Maintainer Adeline Clarke <adelinepclarke@gmail.com>

Repository CRAN

Date/Publication 2025-03-06 16:30:02 UTC

Contents

`child` *Child Barometer 2016 response data*

Description

This data contains background variables and the responses to q3 "Missä asioissa olet hyvä? (Avokysymys)", q7 "Kertoisitko, mitä sinun mielestäsi kiusaaminen on? (Avokysymys)", and q11 "Mikä tekee sinut iloiseksi? (Avokysymys)" in the FSD3134 Lapsibarometri 2016 dataset.

Usage

```
child
```

Format

'child' A dataframe with 414 rows and 8 columns:

fsd_id FSD case id

q3 'Which things are you good at?' response text

q7 'What do you think bullying is?' response text

q11 'What makes you happy?' response text

paino Weight

gender Gender)

major_region Major region)

daycare_before_school Daycare before pre-school

Source

<https://urn.fi/urn:nbn:fi:fsd:T-FSD3134>

`dev_coop` *Young People's Views on Development Cooperation 2012 response data*

Description

This data contains background variables and the responses to q11_1 'Jatka lausetta: Kehitysmaa on maa, jossa... (Avokysymys)', q11_2 'Jatka lausetta: Kehitysyhteistyö on toimintaa, jossa... (Avokysymys)', q11_3' Jatka lausetta: Maailman kolme suurinta ongelmaa ovat... (Avokysymys)' in the FSD2821 Nuorten ajatuksia kehitysyhteistyöstä 2012 dataset.

Usage

```
dev_coop
```

Format

'dev_coop' A dataframe with 925 rows and 9 columns:

fsd_id FSD case id

q11_1 response text for q11_1

q11_2 response text for q11_2

q11_3 response text for q11_3

paino Weight

gender Gender

year_of_birth Year of Birth

region Region of Residence

education_level Education level

Source

<https://urn.fi/urn:nbn:fi:fsd:T-FSD2821>

english_sample_survey *English Sample Survey Data: Patient Joe*

Description

This data contains English text responses to ""Joe's doctor told him that he would need to return in two weeks to find out whether or not his condition had improved. But when Joe asked the receptionist for an appointment, he was told that it would be over a month before the next available appointment. What should Joe do?" as well as categorisation of these responses by two coders as either destructive, passive, somewhat proactive, or proactive.

Usage

```
english_sample_survey
```

Format

'english_sample_survey' A dataframe with 585 rows and 5 columns:

id ID

label Label: destructive, passive, somewhat proactive, or proactive

label_coder1 Label from coder 1

label_coder2 Label from coder 2

text Text of response

Source

<https://doi.org/10.7802/2474>

fst_child *Child Barometer 2016 Bullying response data in CoNLL-U format with NLTK stopwords removed and background variables*

Description

This data contains the responses to q7 "Kertoisitko, mitä sinun mielestäsi kiusaaminen on? (Avokysymys)" in the FSD3134 Lapsibarometri 2016 dataset in CoNLL-U format with NLTK stopwords and punctuation removed plus weights and background variables.

Usage

```
fst_child
```

Format

'fst_child' A dataframe with 1580 rows and 18 columns:

doc_id the identifier of the document

paragraph_id the identifier of the paragraph

sentence_id the identifier of the sentence

sentence the text of the sentence for which this token is part of

token_id Word index, integer starting at 1 for each new sentence; may be a range for multi-word tokens; may be a decimal number for empty nodes.

token Word form or punctuation symbol.

lemma Lemma or stem of word form.

upos Universal part-of-speech tag.

xpos Language-specific part-of-speech tag; underscore if not available.

feats List of morphological features from the universal feature inventory or from a defined language-specific extension; underscore if not available.

head_token_id Head of the current word, which is either a value of token_id or zero (0).

dep_rel Universal dependency relation to the HEAD (root iff HEAD = 0) or a defined language-specific subtype of one.

deps Enhanced dependency graph in the form of a list of head-deprel pairs.

misc Any other annotation.

weight Weight

gender Gender

major_region Major region

daycare_before_school Daycare before pre-school

Source

<https://urn.fi/urn:nbn:fi:fsd:T-FSD3134>

---

fst_child_2 *Child Barometer 2016 Bullying response data in CoNLL-U format with NLTK stopwords removed*

Description

This data contains the responses to q7 "Kertoisitko, mitä sinun mielestäsi kiusaaminen on? (Avokysymys)" in the FSD3134 Lapsibarometri 2016 dataset in CoNLL-U format with NLTK stopwords and punctuation removed.

Usage

```
fst_child_2
```

Format

'fst_child_2' A dataframe with 1580 rows and 14 columns:

doc_id the identifier of the document

paragraph_id the identifier of the paragraph

sentence_id the identifier of the sentence

sentence the text of the sentence for which this token is part of

token_id Word index, integer starting at 1 for each new sentence; may be a range for multi-word tokens; may be a decimal number for empty nodes.

token Word form or punctuation symbol.

lemma Lemma or stem of word form.

upos Universal part-of-speech tag.

xpos Language-specific part-of-speech tag; underscore if not available.

feats List of morphological features from the universal feature inventory or from a defined language-specific extension; underscore if not available.

head_token_id Head of the current word, which is either a value of token_id or zero (0).

dep_rel Universal dependency relation to the HEAD (root iff HEAD = 0) or a defined language-specific subtype of one.

deps Enhanced dependency graph in the form of a list of head-deprel pairs.

misc Any other annotation.

Source

<https://urn.fi/urn:nbn:fi:fsd:T-FSD3134>

---

fst_cn_compare_plot *Concept Network- Plot comparison Concept Network*

Description

Creates a Concept Network plot from a list of edges and nodes (and their respective weights) which indicates unique words in this plot in comparison to another Network.

Usage

```
fst_cn_compare_plot(
  edges,
  nodes,
  concepts,
  unique_lemmas,
  name = NULL,
  concept_colour = "#cd1719",
  unique_colour = "#4DAF4A",
```

```
  min_edge = NULL,
  max_edge = NULL,
  min_node = NULL,
  max_node = NULL,
  title_size = 20
)
```

Arguments

edges	Output of ‘fst_cn_edges()‘, dataframe of ’edges’ connecting two words.
nodes	Output of ‘fst_cn_nodes()‘, dataframe of relevant lemmas and their associated pagerank.
concepts	List of terms which have been searched for, separated by commas.
unique_lemmas	List of unique lemmas, output of ‘fst_cn_get_unique()‘
name	An optional "name" for the plot, default is ‘NULL‘ and a generic title ("TextRank extracted keyword occurrences") will be used.
concept_colour	Colour to display concept words, default is ‘"indianred"‘.
unique_colour	Colour to display unique words, default is ‘"darkgreen"‘.
min_edge	A numeric value for the scale of the edges, the smallest co_occurrence value for an edge across all Networks to be plotted together.
max_edge	A numeric value for the scale of the edges, the largest co_occurrence value for an edge across all Networks to be plotted together.
min_node	A numeric value for the scale of the nodes, the smallest pagerank value for a node across all Networks to be plotted together.
max_node	A numeric value for the scale of the nodes, the largest pagerank value for a node across all Networks to be plotted together.
title_size	size to display plot title

Value

Plot of concept network with concept and unique words (nodes) highlighted.

Examples

```
pos_filter <- c("NOUN", "VERB", "ADJ", "ADV")
e1 <- fst_cn_edges(fst_child, "lyödä", pos_filter = pos_filter)
e2 <- fst_cn_edges(fst_child, "lyöminen", pos_filter = pos_filter)
n1 <- fst_cn_nodes(fst_child, e1)
n2 <- fst_cn_nodes(fst_child, e2)
u <- fst_cn_get_unique_separate(n1, n2)

fst_cn_compare_plot(e1, n1, "lyödä", unique_lemma = u)
fst_cn_compare_plot(e2, n2, "lyöminen", u, unique_colour = "purple")
```

fst_cn_edges *Concept Network - Get TextRank edges*

Description

This function takes a string of terms (separated by commas) or a single term and, using ‘fst_cn_search()‘ find words connected to these searched terms. Then, a dataframe is returned of ’edges’ between two words which are connected together in an frequently-occurring n-gram containing a concept term.

Usage

```
fst_cn_edges(
  data,
  concepts,
  threshold = NULL,
  norm = "number_words",
  pos_filter = NULL
)
```

Arguments

data	A dataframe of text in CoNLL-U format, with optional additional columns.
concepts	List of terms to search for, separated by commas.
threshold	A minimum number of occurrences threshold for ’edge’ between searched term and other word, default is ‘NULL‘. Note, the threshold is applied before normalisation.
norm	The method for normalising the data. Valid settings are ‘"number_words"‘ (the number of words in the responses), ‘"number_resp"‘ (the number of responses), or ‘NULL‘ (raw count returned, default, also used when weights are applied).
pos_filter	List of UPOS tags for inclusion, default is ‘NULL‘ to include all UPOS tags.

Value

Dataframe of co-occurrences between two connected words.

Examples

```
con <- "kiusata, lyöminen"
fst_cn_edges(fst_child, con, pos_filter = c("NOUN", "VERB", "ADJ", "ADV"))
fst_cn_edges(fst_child, con, pos_filter = 'VERB, NOUN')
fst_cn_edges(fst_child, "lyöminen", threshold = 2, norm = "number_resp")
```

`fst_cn_get_unique` *Concept Network- Get unique nodes from a list of top n-grams tables*

Description

Takes at least two tables of nodes and pagerank (output of 'fst_cn_nodes()') and finds nodes unique to one table.

Usage

```
fst_cn_get_unique(list)
```

Arguments

`list`	A list of top nodes

Value

Dataframe of words and whether word is unique or not.

Examples

```
pos_filter <- 'NOUN, VERB, ADJ, ADV'
e1 <- fst_cn_edges(fst_child, "lyödä", pos_filter = pos_filter)
e2 <- fst_cn_edges(fst_child, "lyöminen", pos_filter = pos_filter)
n1 <- fst_cn_nodes(fst_child, e1)
n2 <- fst_cn_nodes(fst_child, e2)
list_of_nodes <- list()
list_of_nodes <- append(list_of_nodes, list(n1))
list_of_nodes <- append(list_of_nodes, list(n2))
fst_cn_get_unique(list_of_nodes)
```

`fst_cn_get_unique_separate` *Concept Network- Get unique nodes from separate top n-grams tables*

Description

Takes at least two tables of nodes and pagerank (output of 'fst_cn_nodes()') and finds nodes unique to one table.

Usage

```
fst_cn_get_unique_separate(table1, table2, ...)
```

Arguments

table1	The first table.
table2	The second table.
...	Any other tables you want to include.

Value

Dataframe of words and whether word is unique or not.

Examples

```
pos_filter <- c("NOUN", "VERB", "ADJ", "ADV")
e1 <- fst_cn_edges(fst_child, "lyödä", pos_filter = pos_filter)
e2 <- fst_cn_edges(fst_child, "lyöminen", pos_filter = pos_filter)
n1 <- fst_cn_nodes(fst_child, e1)
n2 <- fst_cn_nodes(fst_child, e2)
fst_cn_get_unique_separate(n1, n2)
```

---

fst_cn_nodes *Concept Network - Get TextRank nodes*

Description

This function takes a string of terms (separated by commas) or a single term and, using 'textrank_keywords()' from 'textrank' package, filters data based on 'pos_filter' ranks words which are the filtered for those connected to search terms.

Usage

```
fst_cn_nodes(data, edges, pos_filter = NULL)
```

Arguments

data	A dataframe of text in CoNLL-U format, with optional additional columns.
edges	Output of 'fst_cn_edges()', dataframe of co-occurrences between two words.
pos_filter	List of UPOS tags for inclusion, default is 'NULL' to include all UPOS tags.

Value

A dataframe containing relevant lemmas and their associated pagerank.

Examples

```
con <- "kiusata, lyöminen"
cb <- fst_child
edges <- fst_cn_edges(cb, con, pos_filter = c("NOUN", "VERB", "ADJ", "ADV"))
edges2 <- fst_cn_edges(cb, con, pos_filter = 'NOUN, VERB, ADJ, ADV')
fst_cn_nodes(cb, edges, c("NOUN", "VERB", "ADJ", "ADV"))
fst_cn_nodes(cb, edges, 'NOUN, VERB, ADJ, ADV')
```

fst_cn_plot *Plot Concept Network*

Description

Creates a Concept Network plot from a list of edges and nodes (and their respective weights).

Usage

```
fst_cn_plot(edges, nodes, concepts, title = NULL)
```

Arguments

edges	Output of 'fst_cn_edges()', dataframe of 'edges' connecting two words.
nodes	Output of 'fst_cn_nodes()', dataframe of relevant lemmas and their associated pagerank.
concepts	List of terms which have been searched for, separated by commas.
title	Optional title for plot, default is 'NULL' and a generic title ("TextRank extracted keyword occurrences") will be used.

Value

Plot of Concept Network.

Examples

```
con <- "kiusata, lyöminen"
cb <- fst_child
edges <- fst_cn_edges(cb, con, pos_filter = c("NOUN", "VERB", "ADJ", "ADV"))
nodes <- fst_cn_nodes(cb, edges, c("NOUN", "VERB", "ADJ", "ADV"))
fst_cn_plot(edges = edges, nodes = nodes, concepts = con)
```

fst_cn_search *Concept Network - Search TextRank for concepts*

Description

This function takes a string of terms (separated by commas) or a single term and, using 'textrank_keywords()' from 'textrank' package, filters data based on 'pos_filter' and finds words connected to search terms.

Usage

```
fst_cn_search(data, concepts, pos_filter = NULL)
```

Arguments

data	A dataframe of text in CoNLL-U format, with optional additional columns.
concepts	String of terms to search for, separated by commas.
pos_filter	List of UPOS tags for inclusion, default is 'NULL' to include all UPOS tags.

Value

Dataframe of n-grams containing searched terms.

Examples

```
con <- "kiusata, lyöminen, lyödä, potkia"
pf <- c("NOUN", "VERB", "ADJ", "ADV")
pf2 <- "NOUN, VERB, ADJ, ADV"
fst_cn_search(fst_child, concepts = con, pos_filter = pf)
fst_cn_search(fst_child, concepts = con, pos_filter = pf2)
fst_cn_search(fst_child, concepts = con)
```

---

fst_comparison_cloud *Make comparison cloud*

Description

Creates a comparison wordcloud showing words that occur differently between each group. Data is split based on different values in the 'field' column of formatted data. Results will be shown within the plots pane.

Usage

```
fst_comparison_cloud(
  data,
  field,
  pos_filter = NULL,
  max = 100,
  norm = NULL,
  use_svydesign_weights = FALSE,
  use_svydesign_field = FALSE,
  id = "",
  svydesign = NULL,
  use_column_weights = FALSE,
  exclude_nulls = FALSE,
  rename_nulls = "null_data"
)
```

Arguments

data	A dataframe of text in CoNLL-U format with additional ‘field‘ column for splitting data.
field	Column in ‘data‘ used for splitting groups
pos_filter	List of UPOS tags for inclusion, default is ‘NULL‘ which means all word types included.
max	The maximum number of words to display, default is ‘100‘.
norm	The method for normalising the data. Valid settings are ‘"number_words"‘ (the number of words in the responses), ‘"number_resp"‘ (the number of responses), or ‘NULL‘ (raw count returned, default, also used when weights are applied).
use_svydesign_weights	Option to weight words in the wordcloud using weights from a svydesign object containing the raw data, default is ‘FALSE‘
use_svydesign_field	Option to get ‘field‘ for splitting the data from the svydesign object, default is ‘FALSE‘
id	ID column from raw data, required if ‘use_svydesign_weights = TRUE‘ and must match the ‘docid‘ in formatted ‘data‘.
svydesign	A svydesign object which contains the raw data and weights.
use_column_weights	Option to weight words in the wordcloud using weights from formatted data which includes addition ‘weight‘ column, default is ‘FALSE‘
exclude_nulls	Whether to include NULLs in ‘field‘ column, default is ‘FALSE‘
rename_nulls	What to fill NULL values with if ‘exclude_nulls = FALSE‘.

Value

A comparison cloud from wordcloud package.

Examples

```
fst_comparison_cloud(fst_child, 'gender', max = 50)
s <- survey::svydesign(id=~1, weights= ~paino, data = child)
i <- 'fsd_id'
c2 <- fst_child_2
fst_comparison_cloud(c2, 'gender', NULL, 100, NULL, TRUE, TRUE, i, s)
T <- TRUE
fst_comparison_cloud(fst_dev_coop, 'education_level', use_column_weights = T)
pf <- c("NOUN", "VERB", "ADJ", "ADV")
pf2 <- "NOUN, VERB, ADJ, ADV"
fst_comparison_cloud(fst_dev_coop, 'gender', pos_filter = pf)
fst_comparison_cloud(fst_dev_coop, 'gender', pos_filter = pf2)
fst_comparison_cloud(fst_dev_coop, 'gender', norm = 'number_resp')
```

`fst_concept_network` *Concept Network - Make Concept Network plot*

Description

This function takes a string of terms (separated by commas) or a single term and, using ‘textrank_keywords()‘ from ‘textrank‘ package, filters data based on ‘pos_filter‘ and finds words connected to search terms. Then it plots a Concept Network based on the calculated weights of these terms and the frequency of co-occurrences.

Usage

```
fst_concept_network(
  data,
  concepts,
  threshold = NULL,
  norm = "number_words",
  pos_filter = NULL,
  title = NULL
)
```

Arguments

`data`	A dataframe of text in CoNLL-U format, with optional additional columns.
`concepts`	List of terms to search for, separated by commas.
`threshold`	A minimum number of occurrences threshold for 'edge' between searched term and other word, default is ‘NULL‘. Note, the threshold is applied before normalisation.
`norm`	The method for normalising the data. Valid settings are ‘"number_words"‘ (the number of words in the responses), ‘"number_resp"‘ (the number of responses), or ‘NULL‘ (raw count returned, default, also used when weights are applied).
`pos_filter`	List of UPOS tags for inclusion, default is ‘NULL‘ to include all UPOS tags.
`title`	Optional title for plot, default is ‘NULL‘ and a generic title ("TextRank extracted keyword occurrences") will be used.

Value

Plot of Concept Network.

Examples

```
data <- fst_child
con <- "kiusata, lyöminen"
pf <- c("NOUN", "VERB", "ADJ", "ADV")
title <- "Bullying Concept Network"
fst_concept_network(data, concepts = con, pos_filter = pf, title = title)
```

`fst_concept_network_compare` *Concept Network- Compare and plot Concept Network*

Description

This function takes a string of terms (separated by commas) or a single term and, using 'textrank_keywords()' from 'textrank' package, filters data based on 'pos_filter' and finds words connected to search terms for each group. Then it plots a Concept Network for each group based on the calculated weights of these terms and the frequency of co-occurrences, indicating any words that are unique to each group's Network plot.

Usage

```
fst_concept_network_compare(
  data,
  concepts,
  field,
  norm = NULL,
  threshold = NULL,
  pos_filter = NULL,
  use_svydesign_field = FALSE,
  id = "",
  svydesign = NULL,
  exclude_nulls = FALSE,
  rename_nulls = "null_data",
  title_size = 20,
  subtitle_size = 15
)
```

Arguments

`data`	A dataframe of text in CoNLL-U format with additional 'field' column for splitting data.
`concepts`	List of terms to search for, separated by commas.
`field`	Column in 'data' used for splitting groups
`norm`	The method for normalising the data. Valid settings are '"number_words"' (the number of words in the responses, default), '"number_resp"' (the number of responses), or 'NULL' (raw count returned).
`threshold`	A minimum number of occurrences threshold for 'edge' between searched term and other word, default is 'NULL'. Note, the threshold is applied before normalisation.
`pos_filter`	List of UPOS tags for inclusion, default is 'NULL' to include all UPOS tags.
`use_svydesign_field`	Option to get 'field' for splitting the data from a svydesign object, default is 'FALSE'

id	ID column from raw data, required if 'use_svydesign_weights = TRUE' and must match the 'docid' in formatted 'data'.
svydesign	A svydesign object which contains the raw data and weights.
exclude_nulls	Whether to include NULLs in 'field' column, default is 'FALSE'
rename_nulls	What to fill NULL values with if 'exclude_nulls = FALSE'.
title_size	size to display plot title
subtitle_size	size to display title of individual concept network

Value

Multiple concept network plots with concept and unique words highlighted.

Examples

```
con1 <- "lyödä, lyöminen"
fst_concept_network_compare(fst_child, concepts = con1, field = 'gender')
s <- survey::svydesign(id=~1, weights= ~paino, data = child)
c2 <- fst_child_2
i <- 'fsd_id'
fst_concept_network_compare(c2, con1, 'gender', NULL, NULL, NULL, TRUE, i, s)
con2 <- "köyhyys, nälänhätä, sota"
fst_concept_network_compare(fst_dev_coop, con2, 'gender')
```

---

fst_dev_coop *Young People's Views on Development Cooperation 2012 q11_3 response data in CoNLL-U format with NTLK stopwords removed and background variables.*

Description

This data contains the responses to Development Cooperation q11_3 dataset in CoNLL-U format with NLTK stopwords and punctuation removed plus weights and background variables.

Usage

```
fst_dev_coop
```

Format

'fst_dev_coop' A dataframe with 4192 rows and 19 columns:

doc_id the identifier of the document

paragraph_id the identifier of the paragraph

sentence_id the identifier of the sentence

sentence the text of the sentence for which this token is part of

token_id Word index, integer starting at 1 for each new sentence; may be a range for multi-word tokens; may be a decimal number for empty nodes.

token Word form or punctuation symbol.

lemma Lemma or stem of word form.

upos Universal part-of-speech tag.

xpos Language-specific part-of-speech tag; underscore if not available.

feats List of morphological features from the universal feature inventory or from a defined language-specific extension; underscore if not available.

head_token_id Head of the current word, which is either a value of token_id or zero (0).

dep_rel Universal dependency relation to the HEAD (root iff HEAD = 0) or a defined language-specific subtype of one.

deps Enhanced dependency graph in the form of a list of head-deprel pairs.

misc Any other annotation.

weight Weight

gender Gender

year_of_birth Year of Birth

region Region of Residence

Source

<https://urn.fi/urn:nbn:fi:fsd:T-FSD2821>

`fst_dev_coop_2` *Young People's Views on Development Cooperation 2012 q11_3 response data in CoNLL-U format with NTLK stopwords removed*

Description

This data contains the responses to Development Cooperation q11_3 dataset in CoNLL-U format with NLTK stopwords and punctuation removed.

Usage

```
fst_dev_coop_2
```

Format

'fst_dev_coop_2' A dataframe with 4192 rows and 14 columns:

doc_id the identifier of the document

paragraph_id the identifier of the paragraph

sentence_id the identifier of the sentence

sentence the text of the sentence for which this token is part of

token_id Word index, integer starting at 1 for each new sentence; may be a range for multi-word tokens; may be a decimal number for empty nodes.

token Word form or punctuation symbol.

lemma Lemma or stem of word form.

upos Universal part-of-speech tag.

xpos Language-specific part-of-speech tag; underscore if not available.

feats List of morphological features from the universal feature inventory or from a defined language-specific extension; underscore if not available.

head_token_id Head of the current word, which is either a value of token_id or zero (0).

dep_rel Universal dependency relation to the HEAD (root iff HEAD = 0) or a defined language-specific subtype of one.

deps Enhanced dependency graph in the form of a list of head-deprel pairs.

misc Any other annotation.

Source

<https://urn.fi/urn:nbn:fi:fsd:T-FSD2821>

`fst_find_stopwords` *Get available stopwords lists*

Description

Returns a tibble containing all available stopword lists for the language, their contents, and the size of the lists.

Usage

```
fst_find_stopwords(language = "fi")
```

Arguments

`language`	two-letter ISO code of the language for the stopword list

Value

A tibble containing the stopwords lists.

Examples

```
fst_find_stopwords()
fst_find_stopwords(language = 'et')
```

`fst_format` *Annotate open-ended survey responses in into CoNLL-U format*

Description

Creates a dataframe in CoNLL-U format from a dataframe containing text from using the [udpipe] package and a language model plus any additional columns that are included such as ‘weights‘ or columns added through ‘add_cols‘.

Usage

```
fst_format(data, question, id, model = "ftb", weights = NULL, add_cols = NULL)
```

Arguments

`data`	A dataframe of survey responses which contains an open-ended question.
`question`	The column in the dataframe which contains the open-ended question.
`id`	The column in the dataframe which contains the ids for the responses.
`model`	A language model available for [udpipe]. ‘"ftb"‘ (default) or ‘"tdt"‘ are recognised as shorthand for "finnish-ftb" and "finnish-tdt". The full list is available in the [udpipe] documentation or via ‘fst_print_available_models()‘.
`weights`	Optional, the column of the dataframe which contains the respective weights for each response.
`add_cols`	Optional, a column (or columns) from the dataframe which contain other information you’d like to retain (for instance, covariate columnns for splitting the data for comparison plots).

Value

Dataframe of annotated text in CoNLL-U format plus any additional columns.

Examples

```
## Not run:
i <- "fsd_id"
fst_format(data = child, question = "q7", id = i)
fst_format(data = child, question = "q7", id = i, model = "tdt")
fst_format(data = child, question = "q7", id = i, weights="paino")
cols <- c("gender", "major_region", "daycare_before_school")
fst_format(child, question = "q7", id = i, add_cols = cols)
fst_format(child, question = "q7", id = i, add_cols = "gender, major_region")
fst_format(child, question = 'q7', id = i, model = 'swedish-talbanken')
unlink("finnish-ftb-ud-2.5-191206.udpipe")
unlink("finnish-tdt-ud-2.5-191206.udpipe")
unlink("swedish-talkbanken-ud-2.5-191206.udpipe")

## End(Not run)
```

`fst_format_svydesign` *Annotate open-ended survey responses within a ‘svydesign‘ object into CoNLL-U format*

Description

Creates a dataframe in CoNLL-U format from a ‘svydesign‘ object including text using the [udpipe] package and a language model plus weights if these are included in the ‘svydesign‘ object and any columns added through ‘add_cols‘.

Usage

```
fst_format_svydesign(
  svydesign,
  question,
  id,
  model = "ftb",
  use_weights = TRUE,
  add_cols = NULL
)
```

Arguments

`svydesign`	A ‘svydesign‘ object which contains an open-ended question.
`question`	The column in the dataframe which contains the open-ended question.
`id`	The column in the dataframe which contains the ids for the responses.
`model`	A language model available for [udpipe]. ‘"ftb"‘ (default) or ‘"tdt"‘ are recognised as shorthand for "finnish-ftb" and "finnish-tdt". The full list is available in the [udpipe] documentation or via ‘fst_print_available_models()‘.
`use_weights`	Optional, whether to use weights within the ‘svydesign‘
`add_cols`	Optional, a column (or columns) from the dataframe which contain other information you’d like to retain (for instance, dimension columnns for splitting the data for comparison plots).

Value

Dataframe of annotated text in CoNLL-U format plus any additional columns.

Examples

```
## Not run:
i <- "fsd_id"
svy_child <- survey::svydesign(id=~1, weights= ~paino, data = child)
fst_format_svydesign(svy_child, question = 'q7', id = 'fsd_id')
fst_format_svydesign(svy_child, question = 'q7', id = i, use_weights = FALSE)
cols <- c('gender', 'major_region')
```

```
fst_format_svydesign(svy_child, 'q7', 'fsd_id', add_cols = cols)

svy_dev <- survey::svydesign(id = ~1, weights = ~paino, data = dev_coop)
fst_format_svydesign(svy_dev, 'q11_1', 'fsd_id', add_cols = 'gender, region')

fst_format_svydesign(svy_dev, 'q11_2', 'fsd_id', 'finnish-ftb')
unlink("finnish-ftb-ud-2.5-191206.udpipe")
unlink("finnish-tdt-ud-2.5-191206.udpipe")

## End(Not run)
```

`fst_freq` *Find and Plot Top Words*

Description

Creates a plot of the most frequently-occurring words (unigrams) within the data. Optionally, weights can be provided either through a ‘weight‘ column in the formatted data, or from a ‘svydesign‘ object with the raw (preformatted) data.

Usage

```
fst_freq(
  data,
  number = 10,
  norm = NULL,
  pos_filter = NULL,
  strict = TRUE,
  name = NULL,
  use_svydesign_weights = FALSE,
  id = "",
  svydesign = NULL,
  use_column_weights = FALSE
)
```

Arguments

`data`	A dataframe of text in CoNLL-U format, with optional additional columns.
`number`	The number of top words to return, default is ‘10‘.
`norm`	The method for normalising the data. Valid settings are ‘"number_words"‘ (the number of words in the responses, default), ‘"number_resp"‘ (the number of responses), or ‘NULL‘ (raw count returned).
`pos_filter`	List of UPOS tags for inclusion, default is ‘NULL‘ which means all word types included.
`strict`	Whether to strictly cut-off at ‘number‘ (ties are alphabetically ordered), default is ‘TRUE‘.

name An optional "name" to add to the plot title, default is 'NULL'.

use_svydesign_weights Option to weight words in the plot using weights from a 'svydesign' containing the raw data, default is 'FALSE'

id ID column from raw data, required if 'use_svydesign_weights = TRUE' and must match the 'docid' in formatted 'data'.

svydesign A 'svydesign' which contains the raw data and weights, required if 'use_svydesign_weights = TRUE'.

use_column_weights Option to weight words in the plot using weights from formatted data which includes addition 'weight' column, default is 'FALSE'

Value

Plot of top words.

Examples

```
fst_freq(fst_child, number = 12, norm = 'number_resp',  name = "All")
fst_freq(fst_child, use_column_weights = TRUE)
s <- survey::svydesign(id=~1, weights= ~paino, data = child)
i <- 'fsd_id'
fst_freq(fst_child_2, use_svydesign_weights = TRUE, svydesign = s, id = i)
```

---

fst_freq_compare *Compare and plot top words*

Description

Find top and unique top words for different groups of participants. Data is split based on different values in the 'field' column of formatted data. Results will be shown within the plots pane.

Usage

```
fst_freq_compare(
  data,
  field,
  number = 10,
  norm = NULL,
  pos_filter = NULL,
  strict = TRUE,
  use_svydesign_weights = FALSE,
  use_svydesign_field = FALSE,
  id = "",
  svydesign = NULL,
  use_column_weights = FALSE,
  exclude_nulls = FALSE,
```

```
  rename_nulls = "null_data",
  unique_colour = "indianred",
  title_size = 20,
  subtitle_size = 15
)
```

Arguments

data	A dataframe of text in CoNLL-U format with additional 'field' column for splitting data.
field	Column in 'data' used for splitting groups
number	The number of n-grams to return, default is '10'.
norm	The method for normalising the data. Valid settings are '"number_words"' (the number of words in the responses), '"number_resp"' (the number of responses), or 'NULL' (raw count returned, default, also used when weights are applied).
pos_filter	List of UPOS tags for inclusion, default is 'NULL' which means all word types included.
strict	Whether to strictly cut-off at 'number' (ties are alphabetically ordered), default is 'TRUE'.
use_svydesign_weights	Option to weight words in the wordcloud using weights from a svydesign object containing the raw data, default is 'FALSE'
use_svydesign_field	Option to get 'field' for splitting the data from the svydesign object, default is 'FALSE'
id	ID column from raw data, required if 'use_svydesign_weights = TRUE' and must match the 'docid' in formatted 'data'.
svydesign	A svydesign object which contains the raw data and weights.
use_column_weights	Option to weight words in the wordcloud using weights from formatted data which includes addition 'weight' column, default is 'FALSE'
exclude_nulls	Whether to include NULLs in 'field' column, default is 'FALSE'
rename_nulls	What to fill NULL values with if 'exclude_nulls = FALSE'.
unique_colour	Colour to display unique words, default is '"indianred"'.
title_size	size to display plot title
subtitle_size	size to display title of individual top words plot

Value

Plots of most frequent words in the plots pane with unique words highlighted.

Examples

```
fst_freq_compare(fst_child, 'gender', number = 10, norm = "number_resp")
fst_freq_compare(fst_child, 'gender', number = 10, norm = NULL)
s <- survey::svydesign(id=~1, weights= ~paino, data = child)
c2 <- fst_child_2
c <- fst_child
g <- 'gender'
fst_freq_compare(c2, g, 10, NULL, NULL, TRUE, TRUE, TRUE, 'fsd_id', s)
fst_freq_compare(c, g, use_column_weights = TRUE, strict = FALSE)
```

---

fst_freq_plot *Make Top Words plot*

Description

Plots most common words.

Usage

```
fst_freq_plot(table, number = NULL, name = NULL)
```

Arguments

table	Output of ‘fst_freq_table()‘ or ‘fst_ngrams_table()‘.
number	Optional number of n-grams for the title, default is ‘NULL‘.
name	An optional "name" for the plot to add to title, default is ‘NULL‘.

Value

Plot of top words.

Examples

```
pf <- c("NOUN", "VERB", "ADJ", "ADV")
top_words <- fst_freq_table(fst_child, number = 15, pos_filter = pf)
fst_freq_plot(top_words, number = 15, name = "Bullying")
```

`fst_freq_table` *Make Top Words Table*

Description

Creates a table of the most frequently-occurring words (unigrams) within the data. Optionally, weights can be provided either through a ‘weight‘ column in the formatted data, or from a ‘svydesign‘ object with the raw (preformatted) data.

Usage

```
fst_freq_table(
  data,
  number = 10,
  norm = NULL,
  pos_filter = NULL,
  strict = TRUE,
  use_svydesign_weights = FALSE,
  id = "",
  svydesign = NULL,
  use_column_weights = FALSE
)
```

Arguments

`data`	A dataframe of text in CoNLL-U format, with optional additional columns.
`number`	The number of top words to return, default is ‘10‘.
`norm`	The method for normalising the data. Valid settings are ‘"number_words"‘ (the number of words in the responses), ‘"number_resp"‘ (the number of r , or ‘NULL‘ (raw count returned, default, also used when weights are applied).
`pos_filter`	List of UPOS tags for inclusion, default is ‘NULL‘ which means all word types included.
`strict`	Whether to strictly cut-off at ‘number‘ (ties are alphabetically ordered), default is ‘TRUE‘.
`use_svydesign_weights`	Option to weight words in the table using weights from a ‘svydesign‘ containing the raw data, default is ‘FALSE‘
`id`	ID column from raw data, required if ‘use_svydesign_weights = TRUE‘ and must match the ‘docid‘ in formatted ‘data‘.
`svydesign`	A ‘svydesign‘ which contains the raw data and weights, required if ‘use_svydesign_weights = TRUE‘.
`use_column_weights`	Option to weight words in the table using weights from formatted data which includes addition ‘weight‘ column, default is ‘FALSE‘

Value

A table of the most frequently occurring words in the data.

Examples

```
pf <- c("NOUN", "VERB", "ADJ", "ADV")
pf2 <- "NOUN, VERB, ADJ, ADV"
fst_freq_table(fst_child, number = 15, strict = FALSE, pos_filter = pf)
fst_freq_table(fst_child, number = 15, strict = FALSE, pos_filter = pf2)
fst_freq_table(fst_child, norm = 'number_words')
fst_freq_table(fst_child, use_column_weights = TRUE)
c2 <- fst_child_2
s <- survey::svydesign(id=~1, weights= ~paino, data = child)
i <- 'fsd_id'
fst_freq_table(c2, use_svydesign_weights = TRUE, svydesign = s, id = i)
```

---

fst_get_unique_ngrams *Get unique n-grams from a list of top n-grams tables*

Description

Takes a list containing at least two tables of n-grams and frequencies (either output of 'fst_freq_table()' or 'fst_ngrams_table()') and finds n-grams unique to one table.

Usage

```
fst_get_unique_ngrams(list_of_top_ngrams)
```

Arguments

list_of_top_ngrams	A list of top ngrams

Value

Dataframe of words and whether word is unique or not.

Examples

```
top_child <- fst_freq_table(fst_child)
top_dev <- fst_freq_table(fst_dev_coop)
list_of_top_words <- list()
list_of_top_words <- append(list_of_top_words, list(top_child))
list_of_top_words <- append(list_of_top_words, list(top_dev))
fst_get_unique_ngrams(list_of_top_words)
```

fst_get_unique_ngrams_separate

Get unique n-grams from separate top n-grams tables

Description

Takes at least two separate tables of n-grams and frequencies (either output of ‘fst_freq_table()‘ or ‘fst_ngrams_table()‘) and finds n-grams unique to one table.

Usage

```
fst_get_unique_ngrams_separate(table1, table2, ...)
```

Arguments

table1	The first n-grams table.
table2	The second n-grams table.
...	Any other n-grams tables you want to include.

Value

Dataframe of words and whether word is unique or not.

Examples

```
top_child <- fst_freq_table(fst_child)
top_dev <- fst_freq_table(fst_dev_coop)
fst_get_unique_ngrams_separate(top_child, top_dev)
```

fst_join_unique

Merge N-grams table with unique words

Description

Merges list of unique words from ‘fst_get_unique_ngrams()‘ with output of ‘fst_freq_table()‘ or ‘fst_ngrams_table()‘ so that unique words can be displayed on comparison plots.

Usage

```
fst_join_unique(table, unique_table)
```

Arguments

table	Output of ‘fst_freq_table()‘ or ‘fst_ngrams_table()‘.
unique_table	Output of ‘fst_get_unique_ngrams()‘.

Value

A table of top n-grams, frequency, and whether the n-gram is "unique".

Examples

```
top_child <- fst_freq_table(fst_child)
top_dev <- fst_freq_table(fst_dev_coop)
unique_words <- fst_get_unique_ngrams_separate(top_child, top_dev)
fst_join_unique(top_child, unique_words)
fst_join_unique(top_dev, unique_words)
```

---

fst_length_compare *Compare response lengths*

Description

Compare length of text responses for different groups of participants. Data is split based on different values in the ‘field‘ column of formatted data. Results will be shown within the plots pane.

Usage

```
fst_length_compare(
  data,
  field,
  incl_sentences = TRUE,
  exclude_nulls = FALSE,
  rename_nulls = "null_data"
)
```

Arguments

data	A dataframe of text in CoNLL-U format with additional ‘field‘ column for splitting data.
field	Column in ‘data‘ used for splitting groups
incl_sentences	Whether to include sentence data in table, default is ‘TRUE‘.
exclude_nulls	Whether to include NULLs in ‘field‘ column, default is ‘FALSE‘
rename_nulls	What to fill NULL values with if ‘exclude_nulls = FALSE‘.

Value

Dataframe summarising response lengths.

Examples

```
fst_length_compare(fst_child, 'gender')
fst_length_compare(fst_dev_coop, 'education_level', incl_sentences = FALSE)
```

fst_length_summary *Make Length Summary Table*

Description

Creates a table summarising distribution of the length of responses.

Usage

```
fst_length_summary(data, desc = "All responses", incl_sentences = TRUE)
```

Arguments

data	dataframe of text in CoNLL-U format, with optional additional columns.
desc	An optional string describing responses in table, default is ‘"All responses"‘.
incl_sentences	Whether to include sentence data in table, default is ‘TRUE‘.

Value

Table summarising distribution of lengths of responses.

Examples

```
fst_length_summary(fst_child, incl_sentences = FALSE)
fst_length_summary(fst_dev_coop, desc = "Q11_3")
```

fst_ngrams *Find and Plot Top N-grams*

Description

Creates a plot of the most frequently-occurring n-grams within the data. Optionally, weights can be provided either through a ‘weight‘ column in the formatted data, or from a ‘svydesign‘ object with the raw (preformatted) data.

Usage

```
fst_ngrams(
  data,
  number = 10,
  ngrams = 1,
  norm = NULL,
  pos_filter = NULL,
  strict = TRUE,
  name = NULL,
```

```
  use_svydesign_weights = FALSE,
  id = "",
  svydesign = NULL,
  use_column_weights = FALSE
)
```

Arguments

data	A dataframe of text in CoNLL-U format, with optional additional columns.
number	The number of top words to return, default is ‘10‘.
ngrams	The type of n-grams, default is ‘1‘.
norm	The method for normalising the data. Valid settings are ‘"number_words"‘ (the number of words in the responses, default), ‘"number_resp"‘ (the number of responses), or ‘NULL‘ (raw count returned).
pos_filter	List of UPOS tags for inclusion, default is ‘NULL‘ which means all word types included.
strict	Whether to strictly cut-off at ‘number‘ (ties are alphabetically ordered), default is ‘TRUE‘.
name	An optional "name" for the plot to add to title, default is ‘NULL‘.
use_svydesign_weights	Option to weight words in the plot using weights from a ‘svydesign‘ containing the raw data, default is ‘FALSE‘
id	ID column from raw data, required if ‘use_svydesign_weights = TRUE‘ and must match the ‘docid‘ in formatted ‘data‘.
svydesign	A ‘svydesign‘ which contains the raw data and weights, required if ‘use_svydesign_weights = TRUE‘.
use_column_weights	Option to weight words in the plot using weights from formatted data which includes addition ‘weight‘ column, default is ‘FALSE‘

Value

Plot of top n-grams

Examples

```
fst_ngrams(fst_child, 12, ngrams = 2, strict = FALSE, name = "All")
c <- fst_child_2
s <- survey::svydesign(id=~1, weights= ~paino, data = child)
i <- 'fsd_id'
T <- TRUE
fst_ngrams(c, ngrams = 3, use_svydesign_weights = T, svydesign = s, id = i)
```

`fst_ngrams_compare` *Compare and plot top n-grams*

Description

Find top and unique top n-grams for different groups of participants. Data is split based on different values in the ‘field‘ column of formatted data. Results will be shown within the plots pane.

Usage

```
fst_ngrams_compare(
  data,
  field,
  number = 10,
  ngrams = 1,
  norm = NULL,
  pos_filter = NULL,
  strict = TRUE,
  use_svydesign_weights = FALSE,
  use_svydesign_field = FALSE,
  id = "",
  svydesign = NULL,
  use_column_weights = FALSE,
  exclude_nulls = FALSE,
  rename_nulls = "null_data",
  unique_colour = "indianred",
  title_size = 20,
  subtitle_size = 15
)
```

Arguments

`data`	A dataframe of text in CoNLL-U format with additional ‘field‘ column for splitting data.
`field`	Column in ‘data‘ used for splitting groups
`number`	The number of n-grams to return, default is ‘10‘.
`ngrams`	The type of n-grams to return, default is ‘1‘.
`norm`	The method for normalising the data. Valid settings are ‘"number_words"‘ (the number of words in the responses), ‘"number_resp"‘ (the number of responses), or ‘NULL‘ (raw count returned, default, also used when weights are applied).
`pos_filter`	List of UPOS tags for inclusion, default is ‘NULL‘ which means all word types included.
`strict`	Whether to strictly cut-off at ‘number‘ (ties are alphabetically ordered), default is ‘TRUE‘.

use_svydesign_weights
: Option to weight words in the wordcloud using weights from a svydesign object containing the raw data, default is 'FALSE'

use_svydesign_field
: Option to get 'field' for splitting the data from the svydesign object, default is 'FALSE'

id
: ID column from raw data, required if 'use_svydesign_weights = TRUE' and must match the 'docid' in formatted 'data'.

svydesign
: A svydesign object which contains the raw data and weights.

use_column_weights
: Option to weight words in the wordcloud using weights from formatted data which includes addition 'weight' column, default is 'FALSE'

exclude_nulls
: Whether to include NULLs in 'field' column, default is 'FALSE'

rename_nulls
: What to fill NULL values with if 'exclude_nulls = FALSE'.

unique_colour
: Colour to display unique words, default is '"indianred"'.

title_size
: size to display plot title

subtitle_size
: size to display title of individual top ngrams plot

Value

Plots of top n-grams in the plots pane with unique n-grams highlighted.

Examples

```
c <- fst_child
g <- 'gender'
fst_ngrams_compare(c, g, ngrams = 4, number = 10, norm = "number_resp")
fst_ngrams_compare(c, g, ngrams = 2, number = 10, norm = NULL)
s <- survey::svydesign(id=~1, weights= ~paino, data = child)
c2 <- fst_child_2
fst_ngrams_compare(c2, g, 10, 3, NULL, NULL, TRUE, TRUE, TRUE, 'fsd_id', s)
fst_ngrams_compare(c, g, 10, 2, use_column_weights = TRUE, strict = TRUE)
```

---

fst_ngrams_compare_plot

Plot comparison n-grams

Description

Plots frequency n-grams with unique n-grams highlighted.

Usage

```
fst_ngrams_compare_plot(
  table,
  number = 10,
  ngrams = 1,
  unique_colour = "indianred",
  name = NULL,
  override_title = NULL,
  title_size = 20
)
```

Arguments

table	The table of n-grams, output of 'get_unique_ngrams()'.
number	The number of n-grams, default is '10'.
ngrams	The type of n-grams, default is '1'.
unique_colour	Colour to display unique words, default is '"indianred"'.
name	An optional "name" for the plot, default is 'NULL'.
override_title	An optional title to override the automatic one for the plot. Default is 'NULL'. If 'NULL', title of plot will be 'number' "Most Common 'Term'". 'Term' is "Words", "Bigrams", or "N-Grams" where N > 2.
title_size	size to display plot title

Value

Plot of top n-grams with unique terms highlighted.

Examples

```
top_child <- fst_freq_table(fst_child)
top_dev <- fst_freq_table(fst_dev_coop)
unique_words <- fst_get_unique_ngrams_separate(top_child, top_dev)
top_child_u <- fst_join_unique(top_child, unique_words)
top_dev_u <- fst_join_unique(top_dev, unique_words)
fst_ngrams_compare_plot(top_child_u, ngrams = 1, name = "Child")
fst_ngrams_compare_plot(top_dev_u, ngrams = 1, name = "Dev", title_size = 10)
```

---

fst_ngrams_plot *Make N-grams plot*

Description

Plots frequency n-grams.

Usage

```
fst_ngrams_plot(table, number = NULL, ngrams = 1, name = NULL)
```

Arguments

table	Output of ‘fst_get_top_words()‘ or ‘fst_get_top_ngrams()‘.
number	Optional number of n-grams for title, default is ‘NULL‘.
ngrams	The type of n-grams, default is ‘1‘.
name	An optional "name" for the plot to add to title, default is ‘NULL‘.

Value

Plot of top n-grams.

Examples

```
top_bigrams <- fst_ngrams_table(fst_child, ngrams = 2, number = 15)
fst_ngrams_plot(top_bigrams, ngrams = 2, number = 15, name = "Children")
```

---

fst_ngrams_table *Make Top N-grams Table*

Description

Creates a table of the most frequently-occurring n-grams within the data. Optionally, weights can be provided either through a ‘weight‘ column in the formatted data, or from a ‘svydesign‘ object with the raw (preformatted) data.

Usage

```
fst_ngrams_table(
  data,
  number = 10,
  ngrams = 1,
  norm = NULL,
  pos_filter = NULL,
  strict = TRUE,
  use_svydesign_weights = FALSE,
  id = "",
  svydesign = NULL,
  use_column_weights = FALSE
)
```

Arguments

data	A dataframe of text in CoNLL-U format, with optional additional columns.
number	The number of n-grams to return, default is ‘10‘.
ngrams	The type of n-grams to return, default is ‘1‘.

norm — The method for normalising the data. Valid settings are ‘"number_words"‘ (the number of words in the responses), ‘"number_resp"‘ (the number of responses), or ‘NULL‘ (raw count returned, default, also used when weights are applied).

pos_filter — List of UPOS tags for inclusion, default is ‘NULL‘ which means all word types included.

strict — Whether to strictly cut-off at ‘number‘ (ties are alphabetically ordered), default is ‘TRUE‘.

use_svydesign_weights — Option to weight words in the table using weights from a ‘svydesign‘ containing the raw data, default is ‘FALSE‘

id — ID column from raw data, required if ‘use_svydesign_weights = TRUE‘ and must match the ‘docid‘ in formatted ‘data‘.

svydesign — A ‘svydesign‘ which contains the raw data and weights, required if ‘use_svydesign_weights = TRUE‘.

use_column_weights — Option to weight words in the table using weights from formatted data which includes addition ‘weight‘ column, default is ‘FALSE‘

Value

A table of the most frequently occurring n-grams in the data.

Examples

```
pf <- c("NOUN", "VERB", "ADJ", "ADV")
pf2 <- "NOUN, VERB, ADJ, ADV"
fst_ngrams_table(fst_child, norm = NULL)
fst_ngrams_table(fst_child, ngrams = 2, norm = "number_resp")
fst_ngrams_table(fst_child, ngrams = 2, pos_filter = pf)
fst_ngrams_table(fst_child, ngrams = 2, pos_filter = pf2)
c2 <- fst_child_2
s <- survey::svydesign(id=~1, weights= ~paino, data = child)
i <- 'fsd_id'
fst_ngrams_table(c2, use_svydesign_weights = TRUE, svydesign = s, id = i)
fst_ngrams_table(fst_child, use_column_weights = TRUE, ngrams = 3)
```

---

fst_ngrams_table2 *Make Top N-grams Table 2*

Description

Creates a table of the most frequently-occurring n-grams within the data. Optionally, weights can be provided either through a ‘weight‘ column in the formatted data, or from a ‘svydesign‘ object with the raw (preformatted) data. Equivalent to ‘fst_get_top_ngrams‘ but doesn’t print message about ties.

Usage

```
fst_ngrams_table2(
  data,
  number = 10,
  ngrams = 1,
  norm = NULL,
  pos_filter = NULL,
  strict = TRUE,
  use_svydesign_weights = FALSE,
  id = "",
  svydesign = NULL,
  use_column_weights = FALSE
)
```

Arguments

data	A dataframe of text in CoNLL-U format, with optional additional columns.
number	The number of n-grams to return, default is ‘10‘.
ngrams	The type of n-grams to return, default is ‘1‘.
norm	The method for normalising the data. Valid settings are ‘"number_words"‘ (the number of words in the responses, default), ‘"number_resp"‘ (the number of responses), or ‘NULL‘ (raw count returned).
pos_filter	List of UPOS tags for inclusion, default is ‘NULL‘ which means all word types included.
strict	Whether to strictly cut-off at ‘number‘ (ties are alphabetically ordered), default is ‘TRUE‘.
use_svydesign_weights	Option to weight words in the table using weights from a ‘svydesign‘ containing the raw data, default is ‘FALSE‘
id	ID column from raw data, required if ‘use_svydesign_weights = TRUE‘ and must match the ‘docid‘ in formatted ‘data‘.
svydesign	A ‘svydesign‘ which contains the raw data and weights, required if ‘use_svydesign_weights = TRUE‘.
use_column_weights	Option to weight words in the table using weights from formatted data which includes addition ‘weight‘ column, default is ‘FALSE‘

Value

A table of the most frequently occurring n-grams in the data.

Examples

```
fst_ngrams_table2(fst_child, norm = NULL)
fst_ngrams_table2(fst_child, ngrams = 2, norm = "number_resp")
c <- fst_child_2
s <- survey::svydesign(id=~1, weights= ~paino, data = child)
```

```
i <- 'fsd_id'
T <- TRUE
fst_ngrams_table2(c, 10, 2, use_svydesign_weights = T, svydesign = s, id = i)
```

`fst_pos` *Make POS Summary Table*

Description

Creates a summary table for the input CoNLL-U data which counts the number of words of each part-of-speech tag within the data.

Usage

```
fst_pos(data)
```

Arguments

`data`	A dataframe of text in CoNLL-U format, with optional additional columns.

Value

A dataframe with a count and proportion of each UPOS tag in the data and the full name of the tag.

Examples

```
fst_pos(fst_child)
fst_pos(fst_dev_coop)
```

`fst_pos_compare` *Compare parts-of-speech*

Description

Count each POS type for different groups of participants. Data is split based on different values in the ‘field‘ column of formatted data. Results will be shown within the plots pane.

Usage

```
fst_pos_compare(data, field, exclude_nulls = FALSE, rename_nulls = "null_data")
```

Arguments

`data`	A dataframe of text in CoNLL-U format with additional ‘field‘ column for splitting data.
`field`	Column in ‘data‘ used for splitting groups
`exclude_nulls`	Whether to include NULLs in ‘field‘ column, default is ‘FALSE‘
`rename_nulls`	What to fill NULL values with if ‘exclude_nulls = FALSE‘.

Value

Table of POS tag counts for the groups.

Examples

```
fst_pos_compare(fst_child, 'gender')
fst_pos_compare(fst_dev_coop, 'region')
```

`fst_prepare` *Read In and format survey text responses*

Description

Creates a dataframe in CoNLL-U format from a dataframe containing text from using the [udpipe] package and a language model plus any additional columns that are included such as ‘weights‘ or columns added through ‘add_cols‘. Stopwords and punctuation are optionally removed if the the ‘stopword_list‘ argument is not "none".

Usage

```
fst_prepare(
  data,
  question,
  id,
  model = "ftb",
  stopword_list = "nltk",
  language = "fi",
  weights = NULL,
  add_cols = NULL,
  manual = FALSE,
  manual_list = ""
)
```

Arguments

`data`	A dataframe of survey responses which contains an open-ended question.
`question`	The column in the dataframe which contains the open-ended question.
`id`	The column in the dataframe which contains the ids for the responses.
`model`	A language model available for [udpipe]. ‘"ftb"‘ (default) or ‘"tdt"‘ are recognised as shorthand for "finnish-ftb" and "finnish-tdt". The full list is available in the [udpipe] documentation or via ‘fst_print_available_models()‘.
`stopword_list`	A valid stopword list, default is ‘"nltk"‘, ‘"manual"‘ can be used to indicate that a manual list will be provided, or ‘"none"‘ if you don’t want to remove stopwords known as ’source’ in ‘stopwords::stopwords‘
`language`	two-letter ISO code for the language for the stopword list

weights	Optional, the column of the dataframe which contains the respective weights for each response.
add_cols	Optional, a column (or columns) from the dataframe which contain other information you'd like to retain (for instance, dimension columnns for splitting the data for comparison plots).
manual	An optional boolean to indicate that a manual list will be provided, 'stopword_list = "manual"' can also or instead be used.
manual_list	A manual list of stopwords.

Details

'fst_prepare_conllu()' produces a dataframe containing survey text responses in CoNLL-U format with stopwords optionally removed.

Value

A dataframe of text in CoNLL-U format.

Examples

```
## Not run:
i <- "fsd_id"
cb <- child
dev <- dev_coop
fst_prepare(data = cb, question = "q7", id = 'fsd_id', weights = 'paino')
fst_prepare(data = dev, question = "q11_2", id = i, add_cols = c('gender'))
fst_prepare(data = dev, question = "q11_3", id = i, add_cols = 'gender')
fst_prepare(data = child, question = "q7", id = i, model = 'swedish-lines')
unlink("finnish-ftb-ud-2.5-191206.udpipe")
unlink("finnish-tdt-ud-2.5-191206.udpipe")
unlink("swedish-lines-ud-2.5-191206.udpipe")

## End(Not run)
```

---

fst_prepare_svydesign *Read In and format survey text responses from 'svydesign' object*

Description

Creates a dataframe in CoNLL-U format from a 'svydesign' object including text using the [udpipe] package and a language model plus weights if these are included in the 'svydesign' object and any columns added through 'add_cols'.Stopwords and punctuation are optionally removed if the the 'stopword_list' argument is not "none".

Usage

```
fst_prepare_svydesign(
  svydesign,
  question,
  id,
  model = "ftb",
  stopword_list = "nltk",
  language = "fi",
  use_weights = TRUE,
  add_cols = NULL,
  manual = FALSE,
  manual_list = ""
)
```

Arguments

svydesign	A ‘svydesign‘ object which contains an open-ended question.
question	The column in the dataframe which contains the open-ended question.
id	The column in the dataframe which contains the ids for the responses.
model	A language model available for [udpipe]. ‘"ftb"‘ (default) or ‘"tdt"‘ are recognised as shorthand for "finnish-ftb" and "finnish-tdt". The full list is available in the [udpipe] documentation or via ‘fst_print_available_models()‘.
stopword_list	A valid stopword list, default is ‘"nltk"‘, or ‘"none"‘.
language	two-letter ISO code for the language for the stopword list
use_weights	Optional, whether to use weights within the ‘svydesign‘
add_cols	Optional, a column (or columns) from the dataframe which contain other information you’d like to retain (for instance, dimension columnns for splitting the data for comparison plots).
manual	An optional boolean to indicate that a manual list will be provided, ‘stopword_list = "manual"‘ can also or instead be used.
manual_list	A manual list of stopwords.

Details

‘fst_prepare_svydesign()‘ produces a dataframe containing survey text responses in CoNLL-U format with stopwords optionally removed.

Value

A dataframe of text in CoNLL-U format.

Examples

```
## Not run:
i <- "fsd_id"
svy_child <- survey::svydesign(id=~1, weights= ~paino, data = child)
```

```
fst_prepare_svydesign(svy_child, question = "q7", id = i, use_weights = TRUE)

svy_d <- survey::svydesign(id = ~1, weights = ~paino, data =dev_coop)
fst_prepare_svydesign(svy_d, question = "q11_2", id = i, add_cols = 'gender')

fst_prepare_svydesign(svy_d, 'q11_2', i, 'finnish-ftb', 'nltk', 'fi')
unlink("finnish-ftb-ud-2.5-191206.udpipe")
unlink("finnish-tdt-ud-2.5-191206.udpipe")

## End(Not run)
```

`fst_print_available_models` *Find treebanks available for use*

Description

Find treebanks available for use

Usage

```
fst_print_available_models(search = NULL)
```

Arguments

`search`	An optional string for filtering the list, name of language in English, eg. 'estonian'

Value

List of available treebanks, filtered

Examples

```
fst_print_available_models()
fst_print_available_models(search = "swedish")
```

`fst_rm_stop_punct` *Remove stopwords and punctuation from CoNLL-U dataframe*

Description

Removes stopwords and punctuation from a dataframe containing survey text data which is already in CoNLL-U format.

Usage

```
fst_rm_stop_punct(
  data,
  stopword_list = "nltk",
  language = "fi",
  manual = FALSE,
  manual_list = ""
)
```

Arguments

data	A dataframe of text in CoNLL-U format.
stopword_list	A valid stopword list, default is '"nltk"', '"manual"' can be used to indicate that a manual list will be provided, or '"none"' if you don't want to remove stopwords, known as 'source' in 'stopwords::stopwords'
language	two-letter ISO code of the language for the stopword list
manual	An optional boolean to indicate that a manual list will be provided, 'stopword_list = "manual"' can also or instead be used.
manual_list	A manual list of stopwords.

Value

A dataframe of text in CoNLL-U format without stopwords and punctuation.

Examples

```
## Not run:
c <- fst_format(child, question = 'q7', id = 'fsd_id')
fst_rm_stop_punct(c)
fst_rm_stop_punct(c, stopword_list = "snowball")
fst_rm_stop_punct(c, "stopwords-iso")

mlist <- c('en', 'et', 'ei', 'emme', 'ette', 'eivät', 'minä', 'minum')
mlist2 <- "en, et, ei, emme, ette, eivät, minä, minum"
fst_rm_stop_punct(c, manual = TRUE, manual_list = mlist)
fst_rm_stop_punct(c, stopword_list = "manual", manual_list = mlist)
unlink("finnish-ftb-ud-2.5-191206.udpipe")

## End(Not run)
```

---

fst_summarise *Make Summary Table*

Description

Creates a summary table for the input CoNLL-U data which provides the response count and proportion, total number of words, the number of unique words, and the number of unique lemmas.

Usage

```
fst_summarise(data, desc = "All responses")
```

Arguments

data	A dataframe of text in CoNLL-U format, with optional additional columns.
desc	A string describing responses in table, default is ‘"All responses"‘.

Value

A dataframe with summary information for the data including response rate and word counts.

Examples

```
fst_summarise(fst_child)
fst_summarise(fst_dev_coop, "Q11_3")
```

---

fst_summarise_compare *Make comparison summary*

Description

Compare text responses for different groups of participants. Data is split based on different values in the ‘field‘ column of formatted data. Results will be shown within the plots pane.

Usage

```
fst_summarise_compare(
  data,
  field,
  exclude_nulls = FALSE,
  rename_nulls = "null_data"
)
```

Arguments

data	A dataframe of text in CoNLL-U format with additional ‘field‘ column for splitting data.
field	Column in ‘data‘ used for splitting groups
exclude_nulls	Whether to include NULLs in ‘field‘ column, default is ‘FALSE‘
rename_nulls	What to fill NULL values with if ‘exclude_nulls = FALSE‘.

Value

Summary table of responses between groups.

Examples

```
fst_summarise_compare(fst_child, 'gender')
fst_summarise_compare(fst_dev_coop, 'gender')
```

`fst_summarise_short` *Make Simple Summary Table*

Description

Creates a summary table for the input CoNLL-U data which provides the total number of words, the number of unique words, and the number of unique lemmas.

Usage

```
fst_summarise_short(data)
```

Arguments

`data`	A dataframe of text in CoNLL-U format, with optional additional columns.

Value

A dataframe with summary information on word counts for the data.

Examples

```
fst_summarise_short(fst_child)
fst_summarise_short(fst_dev_coop)
```

`fst_use_svydesign` *Add 'svydesign' weights to CoNLL-U data*

Description

This function takes data in CoNLL-U format and a 'svydesign' (from 'survey' package) object with weights in it and merges the weights, and any additional columns into the formatted data.

Usage

```
fst_use_svydesign(data, svydesign, id, add_cols = NULL, add_weights = TRUE)
```

Arguments

data	A dataframe of text in CoNLL-U format, with optional additional columns.
svydesign	A ‘svydesign‘ object containing the raw data which produced the ‘data‘
id	ID column from raw data, must match the ‘docid‘ in formatted ‘data‘
add_cols	Optional, a column (or columns) from the dataframe which contain other information you’d need (for instance, covariate column for splitting the data for comparison plots).
add_weights	Optional, a boolean for whether to add weights from svydesign object, default is ‘TRUE‘.

Value

A dataframe of text in CoNLL-U format plus a ‘’weight’‘ column and optional other columns

Examples

```
svy_child <- survey::svydesign(id=~1, weights= ~paino, data = child)
fst_use_svydesign(data = fst_child_2, svydesign = svy_child, id = 'fsd_id')

svy_dev <- survey::svydesign(id = ~1, weights = ~paino, data = dev_coop)
fst_use_svydesign(data = fst_dev_coop_2, svydesign = svy_dev, id = 'fsd_id')
```

---

fst_wordcloud *Make Wordcloud*

Description

Creates a wordcloud from CoNLL-U data of frequently-occurring words. Optionally, weights can be provided either through a ‘weight‘ column in the formatted data, or from a ‘svydesign‘ object with the raw (preformatted) data.

Usage

```
fst_wordcloud(
  data,
  pos_filter = NULL,
  max = 100,
  use_svydesign_weights = FALSE,
  id = "",
  svydesign = NULL,
  use_column_weights = FALSE
)
```

Arguments

data	A dataframe of text in CoNLL-U format, with optional additional columns.
pos_filter	List of UPOS tags for inclusion, default is 'NULL' which means all word types included.
max	The maximum number of words to display, default is '100'.
use_svydesign_weights	Option to weight words in the wordcloud using weights from a 'svydesign' containing the raw data, default is 'FALSE'
id	ID column from raw data, required if 'use_svydesign_weights = TRUE' and must match the 'docid' in formatted 'data'.
svydesign	A 'svydesign' which contains the raw data and weights, required if 'use_svydesign_weights = TRUE'.
use_column_weights	Option to weight words in the wordcloud using weights from formatted data which includes addition 'weight' column, default is 'FALSE'.

Value

A wordcloud from the data.

Examples

```
fst_wordcloud(fst_child)
fst_wordcloud(fst_child, pos_filter = c("NOUN", "VERB", "ADJ", "ADV"))
fst_wordcloud(fst_child, pos_filter = 'NOUN, VERB, ADJ')
fst_wordcloud(fst_child, use_column_weights = TRUE)
i <- 'fsd_id'
c <- fst_child_2
s <- survey::svydesign(id=~1, weights= ~paino, data = child)
fst_wordcloud(c, use_svydesign_weights = TRUE, id = i, svydesign = s)
```

---

runDemo *Run Shiny App Demo*

Description

Run Shiny App Demo

Usage

```
runDemo()
```

Value

launches the RShiny demo

Examples

```
## Not run:
  runDemo()

## End(Not run)
```

Index