Package 'sweater'

November 7, 2023

Title Speedy Word Embedding Association Test and Extras Using R

Version 0.1.8

Description Conduct various tests for evaluating implicit biases in word embeddings: Word Embedding Association Test (Caliskan et al., 2017), <doi:10.1126/science.aal4230>, Relative Norm Distance (Garg et al., 2018), <doi:10.1073/pnas.1720347115>, Mean Average Cosine Similarity (Mazini et al., 2019) <arXiv:1904.04047>, SemAxis (An et al., 2018) <arXiv:1806.05521>, Relative Negative Sentiment Bias (Sweeney & Najafian, 2019) <doi:10.18653/v1/P19-1162>, and Embedding Coherence Test (Dev & Phillips, 2019) <arXiv:1901.07656>.

License GPL (>= 3)

Encoding UTF-8

LazyData true

RoxygenNote 7.2.3

URL https://github.com/gesistsa/sweater

BugReports https://github.com/gesistsa/sweater/issues

LinkingTo Rcpp

Imports Rcpp, purrr, quanteda, LiblineaR, proxy, data.table, cli, combinat

Suggests covr, testthat (>= 3.0.0)

Config/testthat/edition 3

Depends R (>= 3.5)

NeedsCompilation yes

Author Chung-hong Chan [aut, cre] (<https://orcid.org/0000-0002-6232-7530>)

Maintainer Chung-hong Chan <chainsawtiney@gmail.com>

Repository CRAN

Date/Publication 2023-11-07 16:00:02 UTC

R topics documented:

`calculate_es` *Calculate the effect size of a query*

Description

This function calculates the effect of a query.

Usage

```
calculate_es(x, ...)
```

Arguments

`x`	an S3 object returned from a query, either by the function `query()` or underlying functions such as `mac()`
`...`	additional parameters for the effect size functions • `r` for `weat`: a boolean to denote whether convert the effect size to biserial correlation coefficient. • `standardize` for `weat`: a boolean to denote whether to correct the difference by the standard division. The standardized version can be interpreted the same way as Cohen's d.

Details

The following methods are supported.

- `mac` mean cosine distance value. The value makes sense only for comparison (e.g. before and after debiasing). But a lower value indicates greater association between the target words and the attribute words.
- `rnd` sum of all relative norm distances. It equals to zero when there is no bias.
- `rnsb` Kullback-Leibler divergence of the predicted negative probabilities, P, from the uniform distribution. A lower value indicates less bias.
- `ect` Spearman Coefficient of an Embedding Coherence Test. The value ranges from -1 to +1 and a larger value indicates less bias.
- `weat` The standardized effect size (default) can be interpreted the same way as Cohen's D.

Value

effect size

References

Caliskan, A., Bryson, J. J., & Narayanan, A. (2017). Semantics derived automatically from language corpora contain human-like biases. Science, 356(6334), 183-186. doi:10.1126/science.aal4230

Dev, S., & Phillips, J. (2019, April). Attenuating bias in word vectors. In The 22nd International Conference on Artificial Intelligence and Statistics (pp. 879-887). PMLR.

Garg, N., Schiebinger, L., Jurafsky, D., & Zou, J. (2018). Word embeddings quantify 100 years of gender and ethnic stereotypes. Proceedings of the National Academy of Sciences, 115(16), E3635-E3644. doi:10.1073/pnas.1720347115

Manzini, T., Lim, Y. C., Tsvetkov, Y., & Black, A. W. (2019). Black is to criminal as caucasian is to police: Detecting and removing multiclass bias in word embeddings. arXiv preprint arXiv:1904.04047.

Sweeney, C., & Najafian, M. (2019, July). A transparent framework for evaluating unintended demographic bias in word embeddings. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (pp. 1662-1667).

See Also

`weat_es()`, `mac_es()`, `rnd_es()`, `rnsb_es()`, `ect_es()`

`ect` *Embedding Coherence Test*

Description

This function estimate the Embedding Coherence Test (ECT) of word embeddings (Dev & Philips, 2019). If possible, please use `query()` instead.

Usage

```
ect(w, S_words, A_words, B_words, verbose = FALSE)
```

Arguments

w	a numeric matrix of word embeddings, e.g. from read_word2vec()
S_words	a character vector of the first set of target words. In an example of studying gender stereotype, it can include occupations such as programmer, engineer, scientists...
A_words	a character vector of the first set of attribute words. In an example of studying gender stereotype, it can include words such as man, male, he, his.
B_words	a character vector of the second set of attribute words. In an example of studying gender stereotype, it can include words such as woman, female, she, her.
verbose	logical, whether to display information

Value

A list with class "ect" containing the following components:

- $A_words the input A_words
- $B_words the input B_words
- $S_words the input S_words
- $u_a Cosine similarity between each word vector of S_words and average vector of A_words
- $u_b Cosine similarity between each word vector of S_words and average vector of B_words

References

Dev, S., & Phillips, J. (2019, April). Attenuating bias in word vectors. In The 22nd International Conference on Artificial Intelligence and Statistics (pp. 879-887). PMLR.

See Also

ect_es() can be used to obtain the effect size of the test. plot_ect() can be used to visualize the result.

Examples

```
data(googlenews)
S1 <- c("janitor", "statistician", "midwife", "bailiff", "auctioneer",
"photographer", "geologist", "shoemaker", "athlete", "cashier", "dancer",
"housekeeper", "accountant", "physicist", "gardener", "dentist", "weaver",
"blacksmith", "psychologist", "supervisor", "mathematician", "surveyor",
"tailor", "designer", "economist", "mechanic", "laborer", "postmaster",
"broker", "chemist", "librarian", "attendant", "clerical", "musician",
"porter", "scientist", "carpenter", "sailor", "instructor", "sheriff",
"pilot", "inspector", "mason", "baker", "administrator", "architect",
"collector", "operator", "surgeon", "driver", "painter", "conductor",
"nurse", "cook", "engineer", "retired", "sales", "lawyer", "clergy",
"physician", "farmer", "clerk", "manager", "guard", "artist", "smith",
```

```
"official", "police", "doctor", "professor", "student", "judge",
"teacher", "author", "secretary", "soldier")
A1 <- c("he", "son", "his", "him", "father", "man", "boy", "himself",
"male", "brother", "sons", "fathers", "men", "boys", "males", "brothers",
"uncle", "uncles", "nephew", "nephews")
B1 <- c("she", "daughter", "hers", "her", "mother", "woman", "girl",
"herself", "female", "sister", "daughters", "mothers", "women", "girls",
"females", "sisters", "aunt", "aunts", "niece", "nieces")
garg_f1 <- ect(googlenews, S1, A1, B1)
plot_ect(garg_f1)
```

`ect_es` *Calculate the Spearman Coefficient of an ECT result*

Description

This functions calculates the Spearman Coefficient of an Embedding Coherence Test. The value ranges from -1 to +1 and a larger value indicates less bias. If possible, please use `calculate_es()` instead.

Usage

```
ect_es(x)
```

Arguments

x	an ect object from the `ect()` function.

Value

Spearman Coefficient

References

Dev, S., & Phillips, J. (2019, April). Attenuating bias in word vectors. In The 22nd International Conference on Artificial Intelligence and Statistics (pp. 879-887). PMLR.

glove_math *A subset of the pretrained GLoVE word vectors*

Description

This is a subset of the original pretrained GLoVE word vectors provided by Pennington et al (2017). The same word vectors were used in Caliskan et al. (2017) to study biases.

Usage

```
glove_math
```

Format

An object of class `matrix` (inherits from `array`) with 32 rows and 300 columns.

References

Pennington, J., Socher, R., & Manning, C. D. (2014, October). Glove: Global vectors for word representation. In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP) (pp. 1532-1543).

Caliskan, A., Bryson, J. J., & Narayanan, A. (2017). Semantics derived automatically from language corpora contain human-like biases. Science, 356(6334), 183-186. doi:10.1126/science.aal4230

googlenews *A subset of the pretrained word2vec word vectors*

Description

This is a subset of the original pretrained word2vec word vectors trained on Google News. The same word vectors were used in Garg et al. (2018) to study biases.

Usage

```
googlenews
```

Format

An object of class `matrix` (inherits from `array`) with 116 rows and 300 columns.

References

Garg, N., Schiebinger, L., Jurafsky, D., & Zou, J. (2018). Word embeddings quantify 100 years of gender and ethnic stereotypes. Proceedings of the National Academy of Sciences, 115(16), E3635-E3644. doi:10.1073/pnas.1720347115

mac *Mean average cosine similarity*

Description

This function calculates the mean average cosine similarity (MAC) score proposed in Manzini et al (2019). If possible, please use `query()` instead.

Usage

```
mac(w, S_words, A_words, verbose = FALSE)
```

Arguments

w	a numeric matrix of word embeddings, e.g. from `read_word2vec()`
S_words	a character vector of the first set of target words. In an example of studying gender stereotype, it can include occupations such as programmer, engineer, scientists...
A_words	a character vector of the first set of attribute words. In an example of studying gender stereotype, it can include words such as man, male, he, his.
verbose	logical, whether to display information

Value

A list with class "mac" containing the following components:

- $P a vector of cosine similarity values for every word in S_words
- $S_words the input S_words
- $A_words the input A_words `mac_es()` can be used to obtain the effect size of the test.

References

Manzini, T., Lim, Y. C., Tsvetkov, Y., & Black, A. W. (2019). Black is to criminal as caucasian is to police: Detecting and removing multiclass bias in word embeddings. arXiv preprint arXiv:1904.04047.

Examples

```
data(googlenews)
S1 <- c("janitor", "statistician", "midwife", "bailiff", "auctioneer",
"photographer", "geologist", "shoemaker", "athlete", "cashier", "dancer",
"housekeeper", "accountant", "physicist", "gardener", "dentist", "weaver",
"blacksmith", "psychologist", "supervisor", "mathematician", "surveyor",
"tailor", "designer", "economist", "mechanic", "laborer", "postmaster",
"broker", "chemist", "librarian", "attendant", "clerical", "musician",
"porter", "scientist", "carpenter", "sailor", "instructor", "sheriff",
"pilot", "inspector", "mason", "baker", "administrator", "architect",
"collector", "operator", "surgeon", "driver", "painter", "conductor",
```

```
"nurse", "cook", "engineer", "retired", "sales", "lawyer", "clergy",
"physician", "farmer", "clerk", "manager", "guard", "artist", "smith",
"official", "police", "doctor", "professor", "student", "judge", "teacher",
"author", "secretary", "soldier")
A1 <- c("he", "son", "his", "him", "father", "man", "boy", "himself",
"male", "brother", "sons", "fathers", "men", "boys", "males", "brothers",
"uncle", "uncles", "nephew", "nephews")
x <- mac(googlenews, S1, A1)
x$P
```

`mac_es` *Calculation of MAC Effect Size*

Description

This function calculates the mean of cosine distance values. If possible, please use `calculate_es()` instead.

Usage

```
mac_es(x)
```

Arguments

`x`	an object from the function mac

Value

Mean of all cosine similarity values

Author(s)

Chung-hong Chan

References

Manzini, T., Lim, Y. C., Tsvetkov, Y., & Black, A. W. (2019). Black is to criminal as caucasian is to police: Detecting and removing multiclass bias in word embeddings. arXiv preprint arXiv:1904.04047.

nas *Calculate Normalized Association Score*

Description

This functions quantifies the bias in a set of word embeddings by Caliskan et al (2017). In comparison to WEAT introduced in the same paper, this method is more suitable for continuous ground truth data. See Figure 1 and Figure 2 of the original paper. If possible, please use `query()` instead.

Usage

```
nas(w, S_words, A_words, B_words, verbose = FALSE)
```

Arguments

`w`	a numeric matrix of word embeddings, e.g. from `read_word2vec()`
`S_words`	a character vector of the first set of target words. In an example of studying gender stereotype, it can include occupations such as programmer, engineer, scientists...
`A_words`	a character vector of the first set of attribute words. In an example of studying gender stereotype, it can include words such as man, male, he, his.
`B_words`	a character vector of the second set of attribute words. In an example of studying gender stereotype, it can include words such as woman, female, she, her.
`verbose`	logical, whether to display information

Value

A list with class `"nas"` containing the following components:

- `$P` a vector of normalized association score for every word in S
- `$raw` a list of raw results used for calculating normalized association scores
- `$S_words` the input S_words
- `$A_words` the input A_words
- `$B_words` the input B_words

References

Caliskan, A., Bryson, J. J., & Narayanan, A. (2017). Semantics derived automatically from language corpora contain human-like biases. Science, 356(6334), 183-186. doi:10.1126/science.aal4230

plot_bias *Visualize the bias of words in S*

Description

For `ect`, this function calls `plot_ect()`. For other tests (except `weat`), this function plots the bias of words in `S` as a Cleveland Dot Plot. Plotting the result of `weat` is not supported.

Usage

```
plot_bias(x)

## S3 method for class 'sweater'
plot(x, ...)
```

Arguments

`x`	an S3 object returned from mac, rnd, semaxis, nas or rnsb
`...`	other parameters

Value

a plot

plot_ect *Plot an ECT result on a two-dimensional plane*

Description

This functions plot the words in `S_words` on a 2D plane according to their association with the average vectors of `A_words` and `B_words`. A equality line is also added. Words along the equality line have less bias. Words located on the upper side of the equality line have a stronger association with `A_words` and vice versa.

Usage

```
plot_ect(x, ...)
```

Arguments

`x`	an ect object from the ect function.
`...`	additional parameters to the underlying `plot()` function

Value

a plot

query *A common interface for making query*

Description

This function makes a query based on the supplied parameters. The object can then be displayed by the S3 method `print.sweater()` and plotted by `plot.sweater()`.

Usage

```
query(
  w,
  S_words,
  T_words,
  A_words,
  B_words,
  method = "guess",
  verbose = FALSE,
  ...
)

## S3 method for class 'sweater'
print(x, ...)
```

Arguments

`w`	a numeric matrix of word embeddings, e.g. from `read_word2vec()`
`S_words`	a character vector of the first set of target words. In an example of studying gender stereotype, it can include occupations such as programmer, engineer, scientists...
`T_words`	a character vector of the second set of target words. In an example of studying gender stereotype, it can include occupations such as nurse, teacher, librarian...
`A_words`	a character vector of the first set of attribute words. In an example of studying gender stereotype, it can include words such as man, male, he, his.
`B_words`	a character vector of the second set of attribute words. In an example of studying gender stereotype, it can include words such as woman, female, she, her.
`method`	string, the method to be used to make the query. Available options are: `weat`, `mac`, `nas`, `semaxis`, `rnsb`, `rnd`, `nas`, `ect` and `guess`. If "guess", the function selects one of the following methods based on your provided wordsets. • S_words & A_words - "mac" • S_words, A_words & B_words - "rnd" • S_words, T_words, A_words & B_words - "weat"
`verbose`	logical, whether to display information
`...`	additional parameters for the underlying function

- `l` for "semaxis": an integer indicates the number of words to augment each word in A and B based on cosine , see An et al (2018). Default to 0 (no augmentation).
- `levels` for "rnsb": levels of entries in a hierarchical dictionary that will be applied (see `quanteda::dfm_lookup()`)

x a sweater S3 object

Value

a sweater S3 object

See Also

`weat()`, `mac()`, `nas()`, `semaxis()`, `rnsb()`, `rnd()`, `nas()`, `ect()`

Examples

```
data(googlenews)
S1 <- c("janitor", "statistician", "midwife", "bailiff", "auctioneer",
"photographer", "geologist", "shoemaker", "athlete", "cashier", "dancer",
"housekeeper", "accountant", "physicist", "gardener", "dentist", "weaver",
"blacksmith", "psychologist", "supervisor", "mathematician", "surveyor",
"tailor", "designer", "economist", "mechanic", "laborer", "postmaster",
"broker", "chemist", "librarian", "attendant", "clerical", "musician",
"porter", "scientist", "carpenter", "sailor", "instructor", "sheriff",
"pilot", "inspector", "mason", "baker", "administrator", "architect",
"collector", "operator", "surgeon", "driver", "painter", "conductor",
"nurse", "cook", "engineer", "retired", "sales", "lawyer", "clergy",
"physician", "farmer", "clerk", "manager", "guard", "artist", "smith",
"official", "police", "doctor", "professor", "student", "judge",
"teacher", "author", "secretary", "soldier")
A1 <- c("he", "son", "his", "him", "father", "man", "boy", "himself",
"male", "brother", "sons", "fathers", "men", "boys", "males", "brothers",
"uncle", "uncles", "nephew", "nephews")
B1 <- c("she", "daughter", "hers", "her", "mother", "woman", "girl",
"herself", "female", "sister", "daughters", "mothers", "women", "girls",
"females", "sisters", "aunt", "aunts", "niece", "nieces")
garg_f1 <- query(googlenews, S_words = S1, A_words = A1, B_words = B1)
garg_f1
plot(garg_f1)
```

---

read_word2vec *A helper function for reading word2vec format*

Description

This function reads word2vec text format and return a dense matrix that can be used by this package. The file can have or have not the "verification line", i.e. the first line contains the dimensionality of the matrix. If the verification line exists, the function will check the returned matrix for correctness.

Usage

```
read_word2vec(x)
```

Arguments

x	path to your text file

Value

a dense matrix

---

rnd *Relative Norm Distance*

Description

This function calculate the relative norm distance (RND) of word embeddings. If possible, please use `query()` instead.

Usage

```
rnd(w, S_words, A_words, B_words, verbose = FALSE)
```

Arguments

w	a numeric matrix of word embeddings, e.g. from `read_word2vec()`
S_words	a character vector of the first set of target words. In an example of studying gender stereotype, it can include occupations such as programmer, engineer, scientists...
A_words	a character vector of the first set of attribute words. In an example of studying gender stereotype, it can include words such as man, male, he, his.
B_words	a character vector of the second set of attribute words. In an example of studying gender stereotype, it can include words such as woman, female, she, her.
verbose	logical, whether to display information

Value

A list with class "rnd" containing the following components:

- $norm_diff a vector of relative norm distances for every word in S_words
- $S_words the input S_words
- $A_words the input A_words
- $B_words the input B_words `rnd_es()` can be used to obtain the effect size of the test.

References

Garg, N., Schiebinger, L., Jurafsky, D., & Zou, J. (2018). Word embeddings quantify 100 years of gender and ethnic stereotypes. Proceedings of the National Academy of Sciences, 115(16), E3635-E3644. doi:10.1073/pnas.1720347115

Examples

```
data(googlenews)
S1 <- c("janitor", "statistician", "midwife", "bailiff", "auctioneer",
"photographer", "geologist", "shoemaker", "athlete", "cashier", "dancer",
"housekeeper", "accountant", "physicist", "gardener", "dentist", "weaver",
"blacksmith", "psychologist", "supervisor", "mathematician", "surveyor",
"tailor", "designer", "economist", "mechanic", "laborer", "postmaster",
"broker", "chemist", "librarian", "attendant", "clerical", "musician",
"porter", "scientist", "carpenter", "sailor", "instructor", "sheriff",
"pilot", "inspector", "mason", "baker", "administrator", "architect",
"collector", "operator", "surgeon", "driver", "painter", "conductor",
"nurse", "cook", "engineer", "retired", "sales", "lawyer", "clergy",
"physician", "farmer", "clerk", "manager", "guard", "artist", "smith",
"official", "police", "doctor", "professor", "student", "judge",
"teacher", "author", "secretary", "soldier")
A1 <- c("he", "son", "his", "him", "father", "man", "boy", "himself",
"male", "brother", "sons", "fathers", "men", "boys", "males", "brothers",
"uncle", "uncles", "nephew", "nephews")
B1 <- c("she", "daughter", "hers", "her", "mother", "woman", "girl",
"herself", "female", "sister", "daughters", "mothers", "women", "girls",
"females", "sisters", "aunt", "aunts", "niece", "nieces")
garg_f1 <- rnd(googlenews, S1, A1, B1)
plot_bias(garg_f1)
```

---

rnd_es *Calculation of sum of all relative norm distances*

Description

This function calculates the sum of all relative norm distances from the relative norm distance test. If possible, please use `calculate_es()` instead.

Usage

```
rnd_es(x)
```

Arguments

x	an object from the function rnd

Value

Sum of all relative norm distances

References

Garg, N., Schiebinger, L., Jurafsky, D., & Zou, J. (2018). Word embeddings quantify 100 years of gender and ethnic stereotypes. Proceedings of the National Academy of Sciences, 115(16), E3635-E3644. doi:10.1073/pnas.1720347115

`rnsb` *Relative Negative Sentiment Bias*

Description

This function estimate the Relative Negative Sentiment Bias (RNSB) of word embeddings (Sweeney & Najafian, 2 019). If possible, please use `query()` instead.

Usage

```
rnsb(w, S_words, A_words, B_words, levels = 1, verbose = FALSE)
```

Arguments

`w`	a numeric matrix of word embeddings, e.g. from `read_word2vec()`
`S_words`	a character vector of the first set of target words. In an example of studying gender stereotype, it can include occupations such as programmer, engineer, scientists...
`A_words`	a character vector of the first set of attribute words. In an example of studying gender stereotype, it can include words such as man, male, he, his.
`B_words`	a character vector of the second set of attribute words. In an example of studying gender stereotype, it can include words such as woman, female, she, her.
`levels`	levels of entries in a hierarchical dictionary that will be applied (see `quanteda::dfm_lookup()`)
`verbose`	logical, whether to display information

Value

A list with class `"rnsb"` containing the following components:

- `$classifer` a logistic regression model with L2 regularization trained with LiblineaR
- `$A_words` the input A_words
- `$B_words` the input B_words
- `$S_words` the input S_words
- `$P` the predicted negative sentiment probabilities `rnsb_es()` can be used to obtain the effect size of the test.

References

Sweeney, C., & Najafian, M. (2019, July). A transparent framework for evaluating unintended demographic bias in word embeddings. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (pp. 1662-1667).

Examples

```
data(googlenews)
S1 <- c("janitor", "statistician", "midwife", "bailiff", "auctioneer",
"photographer", "geologist", "shoemaker", "athlete", "cashier", "dancer",
"housekeeper", "accountant", "physicist", "gardener", "dentist", "weaver",
"blacksmith", "psychologist", "supervisor", "mathematician", "surveyor",
"tailor", "designer", "economist", "mechanic", "laborer", "postmaster",
"broker", "chemist", "librarian", "attendant", "clerical", "musician",
"porter", "scientist", "carpenter", "sailor", "instructor", "sheriff",
"pilot", "inspector", "mason", "baker", "administrator", "architect",
"collector", "operator", "surgeon", "driver", "painter", "conductor",
"nurse", "cook", "engineer", "retired", "sales", "lawyer", "clergy",
"physician", "farmer", "clerk", "manager", "guard", "artist", "smith",
"official", "police", "doctor", "professor", "student", "judge",
"teacher", "author", "secretary", "soldier")
A1 <- c("he", "son", "his", "him", "father", "man", "boy", "himself",
"male", "brother", "sons", "fathers", "men", "boys", "males", "brothers",
"uncle", "uncles", "nephew", "nephews")
B1 <- c("she", "daughter", "hers", "her", "mother", "woman", "girl",
"herself", "female", "sister", "daughters", "mothers", "women", "girls",
"females", "sisters", "aunt", "aunts", "niece", "nieces")
garg_f1 <- rnsb(googlenews, S1, A1, B1)
plot_bias(garg_f1)
```

---

rnsb_es *Calculation the Kullback-Leibler divergence*

Description

This function calculates the Kullback-Leibler divergence of the predicted negative probabilities, P, from the uniform distribution. If possible, please use `calculate_es()` instead.

Usage

```
rnsb_es(x)
```

Arguments

x	an rnsb object from the rnsb function.

Value

the Kullback-Leibler divergence.

References

Sweeney, C., & Najafian, M. (2019, July). A transparent framework for evaluating unintended demographic bias in word embeddings. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (pp. 1662-1667).

`semaxis` *Characterise word semantics using the SemAxis framework*

Description

This function calculates the axis and the score using the SemAxis framework proposed in An et al (2018). If possible, please use `query()` instead.

Usage

```
semaxis(w, S_words, A_words, B_words, l = 0, verbose = FALSE)
```

Arguments

`w`	a numeric matrix of word embeddings, e.g. from `read_word2vec()`
`S_words`	a character vector of the first set of target words. In an example of studying gender stereotype, it can include occupations such as programmer, engineer, scientists...
`A_words`	a character vector of the first set of attribute words. In an example of studying gender stereotype, it can include words such as man, male, he, his.
`B_words`	a character vector of the second set of attribute words. In an example of studying gender stereotype, it can include words such as woman, female, she, her.
`l`	an integer indicates the number of words to augment each word in A and B based on cosine , see An et al (2018). Default to 0 (no augmentation).
`verbose`	logical, whether to display information

Value

A list with class `"semaxis"` containing the following components:

- `$P` for each of words in S, the score according to SemAxis
- `$V` the semantic axis vector
- `$S_words` the input S_words
- `$A_words` the input A_words
- `$B_words` the input B_words

References

An, J., Kwak, H., & Ahn, Y. Y. (2018). SemAxis: A lightweight framework to characterize domain-specific word semantics beyond sentiment. arXiv preprint arXiv:1806.05521.

Examples

```
data(glove_math)
S1 <- c("math", "algebra", "geometry", "calculus", "equations",
"computation", "numbers", "addition")
A1 <- c("male", "man", "boy", "brother", "he", "him", "his", "son")
B1 <- c("female", "woman", "girl", "sister", "she", "her", "hers", "daughter")
semaxis(glove_math, S1, A1, B1, l = 0)$P
```

---

small_reddit *A subset of the pretrained word2vec word vectors on Reddit*

Description

This is a subset of the pretrained word2vec word vectors on Reddit provided by An et al. (2018). With this dataset, you can try with the "l" parameter of `semaxis()` up to 10.

Usage

```
small_reddit
```

Format

An object of class `matrix` (inherits from `array`) with 106 rows and 300 columns.

References

An, J., Kwak, H., & Ahn, Y. Y. (2018). SemAxis: A lightweight framework to characterize domain-specific word semantics beyond sentiment. arXiv preprint arXiv:1806.05521.

---

weat *Speedy Word Embedding Association Test*

Description

This functions test the bias in a set of word embeddings using the method by Caliskan et al (2017). If possible, please use `query()` instead.

Usage

```
weat(w, S_words, T_words, A_words, B_words, verbose = FALSE)
```

Arguments

w	a numeric matrix of word embeddings, e.g. from read_word2vec()
S_words	a character vector of the first set of target words. In an example of studying gender stereotype, it can include occupations such as programmer, engineer, scientists...
T_words	a character vector of the second set of target words. In an example of studying gender stereotype, it can include occupations such as nurse, teacher, librarian...
A_words	a character vector of the first set of attribute words. In an example of studying gender stereotype, it can include words such as man, male, he, his.
B_words	a character vector of the second set of attribute words. In an example of studying gender stereotype, it can include words such as woman, female, she, her.
verbose	logical, whether to display information

Value

A list with class "weat" containing the following components:

- $S_diff for each of words in S_words, mean of the mean differences in cosine similarity between words in A_words and words in B_words
- $T_diff for each of words in T_words, mean of the mean differences in cosine similarity between words in A_words and words in B_words
- $S_words the input S_words
- $T_words the input T_words
- $A_words the input A_words
- $B_words the input B_words weat_es() can be used to obtain the effect size of the test; weat_resampling() for a test of significance.

References

Caliskan, A., Bryson, J. J., & Narayanan, A. (2017). Semantics derived automatically from language corpora contain human-like biases. Science, 356(6334), 183-186. doi:10.1126/science.aal4230

Examples

```
# Reproduce the number in Caliskan et al. (2017) - Table 1, "Math vs. Arts"
data(glove_math)
S1 <- c("math", "algebra", "geometry", "calculus", "equations",
"computation", "numbers", "addition")
T1 <- c("poetry", "art", "dance", "literature", "novel", "symphony", "drama", "sculpture")
A1 <- c("male", "man", "boy", "brother", "he", "him", "his", "son")
B1 <- c("female", "woman", "girl", "sister", "she", "her", "hers", "daughter")
sw <- weat(glove_math, S1, T1, A1, B1)
weat_es(sw)
```

`weat_es` *Calculation of WEAT effect size*

Description

This function calculates the effect size from a sweater object. The original implementation in Caliskan et al. (2017) assumes the numbers of words in S and in T must be equal. The current implementation eases this assumption by adjusting the variance with the difference in sample sizes. This adjustment works not so great when the length of S and T are short. It is also possible to convert the Cohen's d to Pearson's correlation coefficient (r). If possible, please use `calculate_es()` instead.

Usage

```
weat_es(x, standardize = TRUE, r = FALSE)
```

Arguments

`x`	an object from the weat function.
`standardize`	a boolean to denote whether to correct the difference by the standard division. The standardized version can be interpreted the same way as Cohen's d.
`r`	a boolean to denote whether convert the effect size to biserial correlation coefficient.

Value

the effect size of the query

References

Caliskan, A., Bryson, J. J., & Narayanan, A. (2017). Semantics derived automatically from language corpora contain human-like biases. Science, 356(6334), 183-186. doi:10.1126/science.aal4230

Examples

```
# Reproduce the number in Caliskan et al. (2017) - Table 1, "Math vs. Arts"
data(glove_math)
S1 <- c("math", "algebra", "geometry", "calculus", "equations",
"computation", "numbers", "addition")
T1 <- c("poetry", "art", "dance", "literature", "novel", "symphony", "drama", "sculpture")
A1 <- c("male", "man", "boy", "brother", "he", "him", "his", "son")
B1 <- c("female", "woman", "girl", "sister", "she", "her", "hers", "daughter")
sw <- weat(glove_math, S1, T1, A1, B1)
weat_es(sw)
```

`weat_exact` *Test of significance for WEAT*

Description

This function conducts the test of significance for WEAT as described in Caliskan et al. (2017). The exact test (proposed in Caliskan et al.) takes an unreasonably long time, if the total number of words in S and T is larger than 10. The resampling test is an approximation of the exact test.

Usage

```
weat_exact(x)

weat_resampling(x, n_resampling = 9999)
```

Arguments

`x`	an object from the weat function.
`n_resampling`	an integer specifying the number of replicates used to estimate the exact test

Value

A list with class `"htest"`

References

Caliskan, A., Bryson, J. J., & Narayanan, A. (2017). Semantics derived automatically from language corpora contain human-like biases. Science, 356(6334), 183-186. doi:10.1126/science.aal4230

Examples

```
# Reproduce the number in Caliskan et al. (2017) - Table 1, "Math vs. Arts"
data(glove_math)
S1 <- c("math", "algebra", "geometry", "calculus", "equations",
"computation", "numbers", "addition")
T1 <- c("poetry", "art", "dance", "literature", "novel", "symphony", "drama", "sculpture")
A1 <- c("male", "man", "boy", "brother", "he", "him", "his", "son")
B1 <- c("female", "woman", "girl", "sister", "she", "her", "hers", "daughter")
sw <- weat(glove_math, S1, T1, A1, B1)
weat_resampling(sw)
```

Index