Package 'qdapDictionaries'

October 13, 2022

Type Package

Title Dictionaries and Word Lists for the 'qdap' Package

Version 1.0.7

Date 2018-03-04

Author Tyler Rinker

Maintainer Tyler Rinker <tyler.rinker@gmail.com>

Depends R (>= 3.0.0)

Imports methods, utils

LazyData TRUE

Description A collection of text analysis dictionaries and word lists for use with the 'qdap' package.

License GPL-2

URL http://trinker.github.com/qdapDictionaries/

BugReports http://github.com/trinker/qdapDictionaries/issues

RoxygenNote 6.0.1

NeedsCompilation no

Repository CRAN

Date/Publication 2018-03-05 11:29:08 UTC

R topics documented:

`abbreviations` *Small Abbreviations Data Set*

Description

A dataset containing abbreviations and their qdap friendly form.

Usage

```
data(abbreviations)
```

Format

A data frame with 14 rows and 2 variables

Details

- abv. Common transcript abbreviations
- rep. qdap representation of those abbreviations

`action.verbs` *Action Word List*

Description

A dataset containing a vector of action words. This is a subset of the Moby project: Moby Part-of-Speech.

Usage

```
data(action.verbs)
```

Format

A vector with 1569 elements

Details

From Grady Ward's Moby project: "This second edition is a particularly thorough revision of the original Moby Part-of-Speech. Beyond the fifteen thousand new entries, many thousand more entries have been scrutinized for correctness and modernity. This is unquestionably the largest P-O-S list in the world. Note that the many included phrases means that parsing algorithms can now tokenize in units larger than a single word, increasing both speed and accuracy."

`adverb` *Adverb Word List*

Description

A dataset containing a vector of adverbs words. This is a subset of the Moby project: Moby Part-of-Speech.

Usage

```
data(adverb)
```

Format

A vector with 13398 elements

Details

From Grady Ward's Moby project: "This second edition is a particularly thorough revision of the original Moby Part-of-Speech. Beyond the fifteen thousand new entries, many thousand more entries have been scrutinized for correctness and modernity. This is unquestionably the largest P-O-S list in the world. Note that the many included phrases means that parsing algorithms can now tokenize in units larger than a single word, increasing both speed and accuracy."

---

amplification.words *Amplifying Words*

Description

A dataset containing a vector of words that amplify word meaning.

Usage

```
data(amplification.words)
```

Format

A vector with 49 elements

Details

Valence shifters are words that alter or intensify the meaning of the polarized words and include negators and amplifiers. Negators are, generally, adverbs that negate sentence meaning; for example the word like in the sentence, "I do like pie.", is given the opposite meaning in the sentence, "I do not like pie.", now containing the negator not. Amplifiers are, generally, adverbs or adjectives that intensify sentence meaning. Using our previous example, the sentiment of the negator altered sentence, "I seriously do not like pie.", is heightened with addition of the amplifier seriously. Whereas de-amplifiers decrease the intensity of a polarized word as in the sentence "I barely like pie"; the word "barely" deamplifies the word like.

BuckleySaltonSWL *Buckley & Salton Stopword List*

Description

A stopword list containing a character vector of stopwords.

Usage

```
data(BuckleySaltonSWL)
```

Format

A character vector with 546 elements

Details

From Onix Text Retrieval Toolkit API Reference: "This stopword list was built by Gerard Salton and Chris Buckley for the experimental SMART information retrieval system at Cornell University. This stopword list is generally considered to be on the larger side and so when it is used, some implementations edit it so that it is better suited for a given domain and audience while others use this stopword list as it stands."

Note

Reduced from the original 571 words to 546.

References

http://www.lextek.com/manuals/onix/stopwords2.html

---

contractions *Contraction Conversions*

Description

A dataset containing common contractions and their expanded form.

Usage

```
data(contractions)
```

Format

A data frame with 70 rows and 2 variables

Details

- contraction. The contraction word.
- expanded. The expanded form of the contraction.

deamplification.words *De-amplifying Words*

Description

A dataset containing a vector of words that de-amplify word meaning.

Usage

```
data(deamplification.words)
```

Format

A vector with 13 elements

Details

Valence shifters are words that alter or intensify the meaning of the polarized words and include negators and amplifiers. Negators are, generally, adverbs that negate sentence meaning; for example the word like in the sentence, "I do like pie.", is given the opposite meaning in the sentence, "I do not like pie.", now containing the negator not. Amplifiers are, generally, adverbs or adjectives that intensify sentence meaning. Using our previous example, the sentiment of the negator altered sentence, "I seriously do not like pie.", is heightened with addition of the amplifier seriously. Whereas de-amplifiers decrease the intensity of a polarized word as in the sentence "I barely like pie"; the word "barely" deamplifies the word like.

DICTIONARY *Nettalk Corpus Syllable Data Set*

Description

A dataset containing syllable counts.

Usage

```
data(DICTIONARY)
```

Format

A data frame with 20137 rows and 2 variables

Details

- word. The word
- syllables. Number of syllables

Note

This data set is based on the Nettalk Corpus but has some researcher word deletions and additions based on the needs of the `syllable_sum` algorithm.

References

Sejnowski, T.J., and Rosenberg, C.R. (1987). "Parallel networks that learn to pronounce English text" in Complex Systems, 1, 145-168. Retrieved from: http://archive.ics.uci.edu/ml/datasets/Connectionist+Bench+(Nettalk+Corpus)

UCI Machine Learning Repository website

---

discourse.markers.alemany

Alemany's Discourse Markers

Description

A dataset containing discourse markers

Usage

```
data(discourse.markers.alemany)
```

Format

A data frame with 97 rows and 5 variables

Details

A dictionary of *discourse markers* from Alemany (2005). "In this lexicon, discourse markers are characterized by their structural (continuation or elaboration) and semantic (revision, cause, equality, context) meanings, and they are also associated to a morphosyntactic class (part of speech, PoS), one of adverbial (A), phrasal (P) or conjunctive (C)... Sometimes a discourse marker is **underspecified** with respect to a meaning. We encode this with a hash. This tends to happen with structural meanings, because these meanings can well be established by discursive mechanisms other than discourse markers, and the presence of the discourse marker just reinforces the relation, whichever it may be." (p. 191).

- marker. The discourse marker
- type. The semantic type (typically overlaps with `semantic` except in the special types
- structural. How the marker is used structurally
- semantic. How the marker is used semantically
- pos. Part of speech: adverbial (A), phrasal (P) or conjunctive (C)

References

Alemany, L. A. (2005). Representing discourse for automatic text summarization via shallow NLP techniques (Unpublished doctoral dissertation). Universitat de Barcelona, Barcelona.

http://www.cs.famaf.unc.edu.ar/~laura/shallowdisc4summ/tesi_electronica.pdf
http://russell.famaf.unc.edu.ar/~laura/shallowdisc4summ/discmar/#description

---

`Dolch` *Dolch List of 220 Common Words*

Description

Edward William Dolch's list of 220 Most Commonly Used Words.

Usage

```
data(Dolch)
```

Format

A vector with 220 elements

Details

Dolch's Word List made up 50-75% of all printed text in 1936.

References

Dolch, E. W. (1936). A basic sight vocabulary. Elementary School Journal, 36, 456-460.

`emoticon` *Emoticons Data Set*

Description

A dataset containing common emoticons (adapted from Popular Emoticon List).

Usage

```
data(emoticon)
```

Format

A data frame with 81 rows and 2 variables

Details

- meaning. The meaning of the emoticon
- emoticon. The graphic representation of the emoticon

References

http://www.lingo2word.com/lists/emoticon_listH.html

`Fry_1000` *Fry's 1000 Most Commonly Used English Words*

Description

A stopword list containing a character vector of stopwords.

Usage

```
data(Fry_1000)
```

Format

A vector with 1000 elements

Details

Fry's 1000 Word List makes up 90% of all printed text.

References

Fry, E. B. (1997). Fry 1000 instant words. Lincolnwood, IL: Contemporary Books.

`function.words` *Function Words*

Description

A vector of function words from John and Muriel Higgins's list used for the text game ECLIPSE. The lest is augmented with additional contractions from `contractions`.

Usage

```
data(function.words)
```

Format

A vector with 350 elements

References

http://myweb.tiscali.co.uk/wordscape/museum/funcword.html

GradyAugmented *Augmented List of Grady Ward's English Words and Mark Kantrowitz's Names List*

Description

A dataset containing a vector of Grady Ward's English words augmented with DICTIONARY, Mark Kantrowitz's names list, other proper nouns, and contractions.

Usage

```
data(GradyAugmented)
```

Format

A vector with 122806 elements

Details

A dataset containing a vector of Grady Ward's English words augmented with proper nouns (U.S. States, Countries, Mark Kantrowitz's Names List, and months) and contractions. That dataset is augmented for spell checking purposes.

References

Moby Thesaurus List by Grady Ward http://www.gutenberg.org

List of names from Mark Kantrowitz http://www.cs.cmu.edu/afs/cs/project/ai-repository/ai/areas/nlp/corpora/names/. A copy of the README is available here per the author's request.

---

interjections *Interjections*

Description

A dataset containing a character vector of common interjections.

Usage

```
data(interjections)
```

Format

A character vector with 139 elements

References

http://www.vidarholen.net/contents/interjections/

`key.pol` *Polarity Lookup Key*

Description

A dataset containing a polarity lookup key (see `polarity`).

Usage

```
data(key.pol)
```

Format

A hash key with words and corresponding values.

References

Hu, M., & Liu, B. (2004). Mining opinion features in customer reviews. National Conference on Artificial Intelligence.

http://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html

`key.power` *Power Lookup Key*

Description

A dataset containing a power lookup key.

Usage

```
data(key.power)
```

Format

A hash key with power words.

References

http://www.wjh.harvard.edu/~inquirer/inqdict.txt

key.strength *Strength Lookup Key*

Description

A dataset containing a strength lookup key.

Usage

```
data(key.strength)
```

Format

A hash key with strength words.

References

http://www.wjh.harvard.edu/~inquirer/inqdict.txt

key.syl *Syllable Lookup Key*

Description

A dataset containing a syllable lookup key (see DICTIONARY).

Usage

```
data(key.syl)
```

Format

A hash key with a modified DICTIONARY data set.

Details

For internal use.

References

UCI Machine Learning Repository website

key.syn *Synonym Lookup Key*

Description

A dataset containing a synonym lookup key.

Usage

```
data(key.syn)
```

Format

A hash key with 10976 rows and 2 variables (words and synonyms).

References

Scraped from: Reverso Online Dictionary. The word list fed to Reverso is the unique words from the combination of `DICTIONARY` and `labMT`.

labMT *Language Assessment by Mechanical Turk (labMT) Sentiment Words*

Description

A dataset containing words, average happiness score (polarity), standard deviations, and rankings.

Usage

```
data(labMT)
```

Format

A data frame with 10222 rows and 8 variables

Details

- word. The word.
- happiness_rank. Happiness ranking of words based on average happiness scores.
- happiness_average. Average happiness score.
- happiness_standard_deviation. Standard deviations of the happiness scores.
- twitter_rank. Twitter ranking of the word.
- google_rank. Google ranking of the word.
- nyt_rank. New York Times ranking of the word.
- lyrics_rank. lyrics ranking of the word.

References

Dodds, P.S., Harris, K.D., Kloumann, I.M., Bliss, C.A., & Danforth, C.M. (2011) Temporal patterns of happiness and information in a global social network: Hedonometrics and twitter. PLoS ONE 6(12): e26752. doi:10.1371/journal.pone.0026752

http://www.plosone.org/article/fetchSingleRepresentation.action?uri=info:doi/10.1371/journal.pone.0026752.s001

Leveled_Dolch *Leveled Dolch List of 220 Common Words*

Description

Edward William Dolch's list of 220 Most Commonly Used Words by reading level.

Usage

```
data(Leveled_Dolch)
```

Format

A data frame with 220 rows and 2 variables

Details

Dolch's Word List made up 50-75% of all printed text in 1936.

- Word. The word
- Level. The reading level of the word

References

Dolch, E. W. (1936). A basic sight vocabulary. Elementary School Journal, 36, 456-460.

NAMES *First Names and Gender (U.S.)*

Description

A dataset containing 1990 U.S. census data on first names.

Usage

```
data(NAMES)
```

Format

A data frame with 5493 rows and 7 variables

Details

- name. A first name.
- per.freq. Frequency in percent of the name by gender.
- cum.freq. Cumulative frequency in percent of the name by gender.
- rank. Rank of the name by gender.
- gender. Gender of the combined male/female list (M/F).
- gender2. Gender of the combined male/female list with "B" in place of overlapping (M/F) names.
- pred.sex. Predicted gender of the names with B's in `gender2` replaced with the gender that had a higher `per.freq`.

References

http://www.census.gov

NAMES_LIST *First Names and Predictive Gender (U.S.) List*

Description

A list version of the `NAMES_SEX` dataset broken down by first letter.

Usage

```
data(NAMES_LIST)
```

Format

A list with 26 elements

Details

Alphabetical list of dataframes with the following variables:

- name. A first name.
- gender2. Gender of the combined male/female list with "B" in place of overlapping (M/F) names.
- pred.sex. Predicted gender of the names with B's in `gender2` replaced with the gender that had a higher `per.freq`.

References

http://www.census.gov

NAMES_SEX *First Names and Predictive Gender (U.S.)*

Description

A truncated version of the NAMES dataset used for predicting.

Usage

```
data(NAMES_SEX)
```

Format

A data frame with 5162 rows and 3 variables

Details

- name. A first name.
- gender2. Gender of the combined male/female list with "B" in place of overlapping (M/F) names.
- pred.sex. Predicted gender of the names with B's in gender2 replaced with the gender that had a higher per.freq.

References

http://www.census.gov

negation.words *Negating Words*

Description

A dataset containing a vector of words that negate word meaning.

Usage

```
data(negation.words)
```

Format

A vector with 23 elements

Details

Valence shifters are words that alter or intensify the meaning of the polarized words and include negators and amplifiers. Negators are, generally, adverbs that negate sentence meaning; for example the word like in the sentence, "I do like pie.", is given the opposite meaning in the sentence, "I do not like pie.", now containing the negator not. Amplifiers are, generally, adverbs or adjectives that intensify sentence meaning. Using our previous example, the sentiment of the negator altered sentence, "I seriously do not like pie.", is heightened with addition of the amplifier seriously. Whereas de-amplifiers decrease the intensity of a polarized word as in the sentence "I barely like pie"; the word "barely" deamplifies the word like.

`negative.words` *Negative Words*

Description

A dataset containing a vector of negative words.

Usage

```
data(negative.words)
```

Format

A vector with 4776 elements

Details

A sentence containing more negative words would be deemed a negative sentence, whereas a sentence containing more positive words would be considered positive.

References

Hu, M., & Liu, B. (2004). Mining opinion features in customer reviews. National Conference on Artificial Intelligence.

http://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html

OnixTxtRetToolkitSWL1 *Onix Text Retrieval Toolkit Stopword List 1*

Description

A stopword list containing a character vector of stopwords.

Usage

```
data(OnixTxtRetToolkitSWL1)
```

Format

A character vector with 404 elements

Details

From Onix Text Retrieval Toolkit API Reference: "This stopword list is probably the most widely used stopword list. It covers a wide number of stopwords without getting too aggressive and including too many words which a user might search upon."

Note

Reduced from the original 429 words to 404.

References

http://www.lextek.com/manuals/onix/stopwords1.html

positive.words *Positive Words*

Description

A dataset containing a vector of positive words.

Usage

```
data(positive.words)
```

Format

A vector with 2003 elements

Details

A sentence containing more negative words would be deemed a negative sentence, whereas a sentence containing more positive words would be considered positive.

References

Hu, M., & Liu, B. (2004). Mining opinion features in customer reviews. National Conference on Artificial Intelligence.

http://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html

---

power.words *Words that Indicate Power*

Description

A subset of the Harvard IV Dictionary containing a vector of words indicating power.

Usage

```
data(power.words)
```

Format

A vector with 624 elements

References

http://www.wjh.harvard.edu/~inquirer/inqdict.txt

---

preposition *Preposition Words*

Description

A dataset containing a vector of common prepositions.

Usage

```
data(preposition)
```

Format

A vector with 162 elements

print.view_data *Prints a view_data Object*

Description

Prints a view_data object.

Usage

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

Arguments

x	The view_data object.
...	ignored

qdapDictionaries *qdapDictionaries*

Description

A collection of dictionaries and Word Lists to Accompany the qdap Package

strong.words *Words that Indicate Strength*

Description

A subset of the Harvard IV Dictionary containing a vector of words indicating strength.

Usage

```
data(strong.words)
```

Format

A vector with 1474 elements

References

http://www.wjh.harvard.edu/~inquirer/inqdict.txt

submit.words *Words that Indicate Submission*

Description

A subset of the Harvard IV Dictionary containing a vector of words indicating submission.

Usage

```
data(submit.words)
```

Format

A vector with 262 elements

References

http://www.wjh.harvard.edu/~inquirer/inqdict.txt

Top100Words *Fry's 100 Most Commonly Used English Words*

Description

A stopword list containing a character vector of stopwords.

Usage

```
data(Top100Words)
```

Format

A character vector with 100 elements

Details

Fry's Word List: The first 25 make up about one-third of all printed material in English. The first 100 make up about one-half of all printed material in English. The first 300 make up about 65% of all printed material in English."

References

Fry, E. B. (1997). Fry 1000 instant words. Lincolnwood, IL: Contemporary Books.

Top200Words — *Fry's 200 Most Commonly Used English Words*

Description

A stopword list containing a character vector of stopwords.

Usage

```
data(Top200Words)
```

Format

A character vector with 200 elements

Details

Fry's Word List: The first 25 make up about one-third of all printed material in English. The first 100 make up about one-half of all printed material in English. The first 300 make up about 65% of all printed material in English."

References

Fry, E. B. (1997). Fry 1000 instant words. Lincolnwood, IL: Contemporary Books.

Top25Words — *Fry's 25 Most Commonly Used English Words*

Description

A stopword list containing a character vector of stopwords.

Usage

```
data(Top25Words)
```

Format

A character vector with 25 elements

Details

Fry's Word List: The first 25 make up about one-third of all printed material in English. The first 100 make up about one-half of all printed material in English. The first 300 make up about 65% of all printed material in English."

References

Fry, E. B. (1997). Fry 1000 instant words. Lincolnwood, IL: Contemporary Books.

`view_data` *List all data sets available in a* **qdapDictionaries**

Description

Lists and describes all the data sets available in **qdapDictionaries**.

Usage

```
view_data(package = "qdapDictionaries")
```

Arguments

`package`	The name of the package.

Value

Returns the data sets of **qdapDictionaries** as a dataframe.

See Also

`data`

Examples

```
view_data()
```

`weak.words` *Words that Indicate Weakness*

Description

A subset of the Harvard IV Dictionary containing a vector of words indicating weakness.

Usage

```
data(weak.words)
```

Format

A vector with 647 elements

References

http://www.wjh.harvard.edu/~inquirer/inqdict.txt

Index