Package: vietnameseConverter (via r-universe)

September 13, 2024

Type Package

Title Convert Vietnamese Encodings

Version 0.4.0

Maintainer Juergen Niedballa <niedballa@izw-berlin.de>

Description Conversion of characters from unsupported Vietnamese character encodings to Unicode characters. These Vietnamese encodings (TCVN3, VISCII, VPS) are not natively supported in R and lead to printing of wrong characters and garbled text (mojibake). This package fixes that problem and provides readable output with the correct Unicode characters (with or without diacritics).

License GPL (>= 2)

Encoding UTF-8

Imports gsubfn, methods, sf, utf8

Depends R (>= 3.5.0)

Suggests testthat, R.rsp, DT

URL https://github.com/jniedballa/vietnameseConverter

VignetteBuilder R.rsp

RoxygenNote 7.1.2

Repository https://ecodynizw.r-universe.dev

RemoteUrl https://github.com/ecodynizw/vietnameseconverter

RemoteRef HEAD

RemoteSha d465ed3b5be1022d67a9fbf9facedecc836c279f

Contents

vietnameseConverter-package 2
decodeVN 2
loadEncodingTableVN 4
vn_samples 5

Index 7

`vietnameseConverter-package` *The vietnameseConverter package*

Description

This package helps you read and use data from Vietnamese sources in R. These data often use Vietnamese legacy character encodings such as TCVN (Vietnam Standards / Tieu chuan Viet Nam) which are still in use today, but are not read correctly in R.

To correct this problem and make such data available in R, this package converts character vectors and entire data frames from legacy Vietnamese encodings to the correct Unicode charactes. It currently supports conversions from three different Vietnamese encodings (TCVN3, VISCII, VPS) to Unicode and vice versa.

The main function is `decodeVN`.

Author(s)

Juergen Niedballa

`decodeVN` *Convert characters from legacy Vietnamese encodings to UTF-8 encoding*

Description

Convert characters from legacy Vietnamese encodings to UTF-8 encoding

Usage

```
decodeVN(
  x,
  from = c("TCVN3", "VISCII", "VPS", "Unicode"),
  to = c("Unicode", "TCVN3", "VISCII", "VPS"),
  diacritics = TRUE
)
```

Arguments

`x`	data.frame, sf object, or character vector
`from`	Text encoding of input x
`to`	Text encoding of output
`diacritics`	logical. Preserve diacritics (TRUE) or not (FALSE)?

Details

Many characters in legacy Vietnamese encodings (e.g. TCVN3, VPS, VISCII) are not read correctly in R, particularly those with diacritics (accents). The particular encodings don't seem to be supported by R, at least on many locales. When R reads them as if they have UTF-8 encoding, it will result in wrong characters being printed and garbled text (Mojibake - see vignette and examples below).

This functions converts character vectors to from various Vietnamese legacy encodings to readable Unicode characters in UTF-8 encoding. By default the function attempts the conversion from TCVN3 to Unicode while preserving the diacritics, but also supports other Vietnamese encodings (TCVN3, VPS, VISCII - via argument `from`). Currently VNI and VNU are not supported.

It works on data frames, spatial objects (from the sf package), and character vectors.

`diacritics = TRUE` will return characters with their diacritics. With `diacritics = FALSE`, the output will be ASCII letters without diacritics. Upper/lower case will be preserved regardless.

The internal search and replace is performed by the `gsubfn` function from the **gsubfn** package. It performs simple character replacements to fix the text.

Currently the function converts from the Vietnamese encodings to Unicode, not vice versa. Please contact the maintainer if the conversion from Unicode to Vietnamese encodings would be relevant for you.

The character conversion table was adapted from http://vietunicode.sourceforge.net/charset/.

Value

character string or data frame (depending on x)

Warning

When printing a data frame with Unicode characters using the standard print method, the R console will show the Unicode escape characters (e.g. "<U+1EA3>") instead of the actual Unicode characters. This is a limitation of the R console. The data are correct and will show correctly when using e.g. View() or when printing columns as vectors.

Examples

```
# First we produce the wrongly formatted character string
# using Unicode symbols is only necessary to create a portable example in the R package
# you don't need to use Unicode characters like this in your data

string <- c("Qu\u00B6ng Tr\u00DE", "An \u00A7\u00ABn", "Th\u00F5a Thi\u00AAn Hu\u00D5")

# Below we have a look at the wrongly formatted character string.
# This is what it would look like when you load TCVN3 encoded data as UTF8
string

# convert character vector from TCVN3 > UTF-8
decodeVN(string)
decodeVN(string, diacritics = FALSE)

# # convert data frame columns from TCVN3 > UTF-8
```

```
df <- data.frame(id = c(1,2,3),
                 name  = string)

df_decode <- decodeVN(df)
df_decode
# NOTE: some characters may be displayed as unicode in the R console
# check the individual column to see if they are correct:
df_decode[,2]

decodeVN(df, diacritics = FALSE)

# using the built-in sample data
data(vn_samples)
decodeVN(vn_samples$TCVN3)   # TCVN -> Unicode   # TCVN3 -> Unicode
decodeVN(vn_samples$TCVN3, diacritics = FALSE)  # TCVN3 -> Unicode (ASCII characters only)
decodeVN(vn_samples$VISCII, from = "VISCII")     # VISCII -> Unicode


# Demonstration for sf object

# create sf object (just for demonstration)
require(sf)
df_geom <- st_sfc(st_point(c(3,4)), st_point(c(10,11)), st_point(c(15,13)))
df_spatial <- st_set_geometry(df, df_geom)

# convert Vietnamese characters
df_spatial_decode <- decodeVN(df_spatial)

df_spatial_decode
df_spatial_decode$name
```

`loadEncodingTableVN` *Load conversion table for Vietnamese characters*

Description

Load conversion table for Vietnamese characters

Usage

```
loadEncodingTableVN(version)
```

Arguments

`version`	Version of this table (1 or 2) - 2 is currentlty being used

Details

The table was adapted from http://vietunicode.sourceforge.net/charset/

Value

data.frame

`vn_samples` *Sample data frames in legacy Vietnamese encodings*

Description

A list with four data frames. The data frames list the provinces of Viet Nam.

The first list item ($Unicode) shows the correct entries. The other three data frames show what loading data encoded in three different Vietnamese encodings would look like when loaded in R.

The list items are:

- **Unicode** - Data frame with correct Unicode characters (reference)
- **TCVN3** - Data frame with TCVN3-encoded characters
- **VISCII** - Data frame with VISCII-encoded characters
- **VPS** - Data frame with VPS-encoded characters

Note that the last 3 are not actually encoded in their respective Vietnamese encodings. Instead, they show what a table in those encodings would look like when loaded into R (or more generally, a system that is not aware of the encodings).

Usage

```
data(vn_samples)
```

Format

A list with 4 data frames

Details

Each data frame contains 5 colums and 63 rows. The first two are character, the last three numeric.

- **Province_city** - Name of province
- **Administrative_center** - Administrative center of the province
- **Area_km2** - Area in km^2
- **Density_perkm2** - Population density (km^-2)
- **HDI_2012** - Human development index in 2012

The first two columns are character, the last three numeric. Only the character columns will be modified by calling `decodeVN`, while the numeric columns will not be changed.

Factors are not converted. If your data frame contains factors, convert these to character first.

Note

The data frame is based on the table of provinces of Viet nam on Wikipedia https://en.wikipedia.org/wiki/Provinces_of_Vietnam with minor edits. The legacy Vietnamese encodings were simulated using the decodeVN function and checked with this online conversion tool: http://www.enderminh.com/minh/vnconversions.aspx.

Index

∗ **datasets**
vn_samples, 5

decodeVN, *2*, 2, *5*, *6*

gsubfn, *3*

loadEncodingTableVN, 4

vietnameseConverter
*(*vietnameseConverter-package*)*, 2
vietnameseConverter-package, 2
vn_samples, 5