NAME

unicharset − character properties file used by tesseract(1)

DESCRIPTION

Tesseract's unicharset file contains information on each symbol (unichar) the Tesseract OCR engine is trained to recognize.

A unicharset file (i.e. *eng.unicharset*) is distributed as part of a Tesseract language pack (i.e. *eng.traineddata*). For information on extracting the unicharset file, see combine_tessdata(1).

The first line of a unicharset file contains the number of unichars in the file. After this line, each subsequent line provides information for a single unichar. The first such line contains a placeholder reserved for the space character. Each unichar is referred to within Tesseract by its Unichar ID, which is the line number (minus 1) within the unicharset file. Therefore, space gets unichar 0.

Each unichar line in the unicharset file (v2+) may have four space−separated fields:

'character' 'properties' 'script' 'id'

Starting with Tesseract v3.02, more information may be given for each unichar:

'character' 'properties' 'glyph_metrics' 'script' 'other_case' 'direction' 'mirror' 'normed_form'

Entries:

character
: The UTF−8 encoded string to be produced for this unichar.

properties
: An integer mask of character properties, one per bit. From least to most significant bit, these are: isalpha, islower, isupper, isdigit, ispunctuation.

glyph_metrics
: Ten comma−separated integers representing various standards for where this glyph is to be found within a baseline−normalized coordinate system where 128 is normalized to x−height.

- min_bottom, max_bottom: the ranges where the bottom of the character can be found.
- min_top, max_top: the ranges where the top of the character may be found.
- min_width, max_width: horizontal width of the character.
- min_bearing, max_bearing: how far from the usual start position does the leftmost part of the character begin.
- min_advance, max_advance: how far from the printer's cell left do we advance to begin the next character.

script
: Name of the script (Latin, Common, Greek, Cyrillic, Han, null).

other_case
: The Unichar ID of the other case version of this character (upper or lower).

direction
: The Unicode BiDi direction of this character, as defined by ICU's enum UCharDirection. (0 = Left to Right, 1 = Right to Left, 2 = European Number...)

mirror
: The Unichar ID of the BiDirectional mirror of this character. For example the mirror of open paren is close paren, but Latin Capital C has no mirror, so it remains a Latin Capital C.

normed_form

The UTF−8 representation of a "normalized form" of this unichar for the purpose of blaming a module for errors given ground truth text. For instance, a left or right single quote may normalize to an ASCII quote.

EXAMPLE (V2)

```
; 10 Common 46
b 3 Latin 59
W 5 Latin 40
7 8 Common 66
= 0 Common 93
```

";" is a punctuation character. Its properties are thus represented by the binary number 10000 (10 in hexadecimal).

"b" is an alphabetic character and a lower case character. Its properties are thus represented by the binary number 00011 (3 in hexadecimal).

"W" is an alphabetic character and an upper case character. Its properties are thus represented by the binary number 00101 (5 in hexadecimal).

"7" is just a digit. Its properties are thus represented by the binary number 01000 (8 in hexadecimal).

"=" is not punctuation nor a digit nor an alphabetic character. Its properties are thus represented by the binary number 00000 (0 in hexadecimal).

Japanese or Chinese alphabetic character properties are represented by the binary number 00001 (1 in hexadecimal): they are alphabetic, but neither upper nor lower case.

EXAMPLE (V3.02)

```
110
NULL 0 NULL 0
N 5 59,68,216,255,87,236,0,27,104,227 Latin 11 0 1 N
Y 5 59,68,216,255,91,205,0,47,91,223 Latin 33 0 2 Y
1 8 59,69,203,255,45,128,0,66,74,173 Common 3 2 3 1
9 8 18,66,203,255,89,156,0,39,104,173 Common 4 2 4 9
a 3 58,65,186,198,85,164,0,26,97,185 Latin 56 0 5 a
. . .
```

CAVEATS

Although the unicharset reader maintains the ability to read unicharsets of older formats and will assign default values to missing fields, the accuracy will be degraded.

Further, most other data files are indexed by the unicharset file, so changing it without re−generating the others is likely to have dire consequences.

HISTORY

The unicharset format first appeared with Tesseract 2.00, which was the first version to support languages other than English. The unicharset file contained only the first two fields, and the "ispunctuation" property was absent (punctuation was regarded as "0", as "=" is in the above example.

SEE ALSO

tesseract(1), combine_tessdata(1), unicharset_extractor(1)

https://github.com/tesseract−ocr/tesseract/wiki/TrainingTesseract

AUTHOR

The Tesseract OCR engine was written by Ray Smith and his research groups at Hewlett Packard (1985−1995) and Google (2006−present).