January 22, 2019

# Recognizing a natural language or language class in source code files

Armijn Hemel

# Recognizing a natural language or language class in source code files

## Abstract

When doing analysis of source code archives from an unknown origin it can be helpful to find out where the code originated from geographically. Comments in these files can be helpful, as they are quite often written in the native natural language of the developer. Finding out which language the file is in can help understanding the flow of the code (example: translating comments) and provenance.

By analyzing the contents of a file and seeing which character sets the contents belong to a better guess can be made.

## Keywords

source code, provenance, unicode, source code analysis

## Background

When doing analysis of unknown source code base from an unknown source it can be beneficial to find out where the person who wrote or changed the code is from. Examples of why could be:

- supply chain management: several companies from different countries (example: Korea, China and Germany) are involved in creating a software product and an error was discovered. By checking the source code to see in which language parts of the code was written or documented it can be easier to find the culprit: it is unlikely that the Chinese or Korean company will have comments in German and vice versa.
- understanding code: if code has been documented in a certain language it makes it easier to find a native speaker of said language to help understand the code.
- refactoring: if a large code base has comments or messages in a certain language but it is intended for an international audience it makes it easier to find the right places and translate the comments and/or localize the messages. This is what the LibreOffice project had to do when they inherited a largely German language code base and needed to prepare it for an international audience[1]

## Method

The following method can be used to find out possible languages or language families in a file with source code. This assumes that the file is in UTF-8 encoding, which might not always be the case and that the programming language used can process UTF-8 encoded files. An example of such a language is Python 3.

1. create a list, dictionary or set to store which characters were found. If the objective is to store how often each character is found a counter data structure can be used.
2. for each line in the file store which characters are used, optionally with a count
3. for each character found check in which Unicode code block the character can be found. Most Unicode code blocks correspond to one or more languages.
4. report the languages found

As an optimization only lines containing comments could be looked at.

## Example code

The following code in Python 3 processes a file in UTF-8 encoding and records for a few character ranges if they were found.

```
#! /usr/bin/python3

# Written by Armijn Hemel 2019
# CC0 1.0 Universal (CC0 1.0)
# Public Domain Dedication

import sys
import collections

# create a counter for each of the characters
charcounter = collections.Counter()

# open the source code file
testfile = open(sys.argv[1])
for i in testfile:
    charcounter.update(i)

# a set of language family names
characters = set()

# process all the characters and check in which Unicode
# range the characters are from and record as such
for i in (charcounter.most_common()):
    # check for Hangul (Korean) characters)
    # https://en.wikipedia.org/wiki/Hangul_Syllables
    if ord(i[0]) >= 0xac00 and ord(i[0]) <= 0xd7af:
        characters.add('hangul')
    # Latin
    elif ord(i[0]) >= 0x41 and ord(i[0]) <= 0x5a:
        characters.add('latin')
    elif ord(i[0]) >= 0x61 and ord(i[0]) <= 0x7a:
        characters.add('latin')
    # Latin-1
    elif ord(i[0]) >= 0xa0 and ord(i[0]) <= 0xff:
        characters.add('latin-1')

# print all the character ranges found
for c in characters:
    print(c)
```

## References

[1] LibreOffice, script to find German comments in LibreOffice's source code: https://github.com/LibreOffice/core/blob/624f84c07ee798f6bbf4381c26262d5d2774a1ed/bin/find-german-comments