

Software Specification

for

Textweiser

A software to classify text

Covers version 1.1.0



Contents

1	Overview on the Software	3
1.1	User's Requirements	3
1.2	Operating Environment	3
1.2.1	Databases	3
1.2.2	Operating Systems and Architectures	3
1.3	Dependencies	4
1.3.1	SQLite	4
1.3.2	Microsoft SQL Server 2008	4
1.4	Use of Resources	4
1.5	Deliveries	4
2	Scope of Service	5
2.1	User Interface	5
2.2	Input	6
2.3	Return Value and Results	6
2.4	Error Handling	6
2.5	Security Considerations	6
2.6	Thread Safety	7
3	Restrictions	7
4	References	7
5	License Informationen	8

This document provides the software specification for **Textweiser**, version 1.1.0.

Textweiser is a C/C++ library that assigns documents to predefined categories (text classifier).

The specification starts with a general introduction to the software and provides a detailed description on the library's scope of service.

1 Overview on the Software

1.1 User's Requirements

Every user with basic knowledge of C/C++ programming and library usage is able to use **Textweiser** right away. In order to use the provided applications, a user must have at least basic knowledge of using the commandline.

With any database except for *SQLite* the user has to have access to the database used.

Textweiser's field of application is generally the field of document management.

1.2 Operating Environment

1.2.1 Databases

Textweiser's standard version is available in variants for the following databases:

- SQLite
- Microsoft SQL Server 2008

A fully configured database server has to be available in the operating environment unless *SQLite* is used.

The Microsoft SQL Server variant of **Textweiser** is only available for Microsoft Windows.

A separate database has to be reserved for **Textweiser**. The data stored within this database must not be altered by users and/or third party software. For a maximum of security, a separate (database-) user should be created for **Textweiser**.

1.2.2 Operating Systems and Architectures

Textweiser's standard version is available for the following operating systems:

- Debian GNU/Linux (x86/x86_64): *Lenny (5.0), Squeeze (6.0)*
- Ubuntu GNU/Linux (x86/x86_64): *LTS (10.04)*
- Red Hat/CentOS GNU/Linux (x86/x86_64): *RHEL 5*
- FreeBSD (x86/x86_64): *7, 8*
- Microsoft Windows (x86): *XP, Server 2003, Vista, Server 2008, 7*
- Microsoft Windows (x86_64): *7, Server 2008 R2*

The software may very well work on other versions and/or distributions without modification although **Textweiser** is only supported in the environments specified above.

Versions for other distributions and/or operating systems may be made available upon request.

1.3 Dependencies

Textweiser has the following dependencies, independent of the database used:

- the system's standard C library
- the system's standard thread library

1.3.1 SQLite

No further dependencies arise, because the database software is already included in **Textweiser**.

1.3.2 Microsoft SQL Server 2008

- Microsoft SQL Server 2008 / 2008 R2
- Standard ODBC library (odbc32.dll)
- SQL Server Native Client (10.0) library

1.4 Use of Resources

The necessary amount of memory (RAM) operatively depends on the size and type of the input, the number of categories, the mode of operation and the database used. It can therefore not be given precisely.

1.5 Deliveries

A complete installation of the software package contains the compiled library, all associated header files and commandline applications. Additionally it includes English man pages and the user manual in an English and German version. Besides that, the source code of example applications is included.

The installation requires – dependent on the system and the database used – about 5 MiB of disk space.

2 Scope of Service

The library assigns text documents to user-defined categories. The categories have to be added first and trained with a set of at least ten representative documents for each category. The profiles of all categories are stored in a database. After training is completed, **Textweiser** can classify documents automatically.

The text is preprocessed with the following computational linguistic steps:

- language identification
- segmentation / tokenization
- phrase recognition
- stop word filtering
- stemming

The library provides the following functionality:

- create/erase database (structures)
- add/rename/delete categories
- obtain a list of all categories
- train/learn documents
- unlearn documents
- classify documents
 - the classification is given with a score (probability)
 - the number of results is user-definable
- optimize profiles of all categories
- create database backup and restore data
- error handling and memory management

Textweiser can handle both flat and mono-hierarchical category structures ("taxonomies"). The automated handling of hierarchies is fully supported. For example when learning a document for a sub-level category, the data is associated with all affected top-level categories as well.

Any further processing of the classification results is up to the application that uses **Textweiser** (for example: automated classification, list suggestions for manual tagging, ...).

2.1 User Interface

All of **Textweiser**'s functionality can be accessed using library functions. Additionally, there are commandline applications provided.

Besides core functions (like adding and training categories and classify texts) both interfaces cover functions to maintain the database. With the commandline interface an easy and automated maintenance of the data is possible, i.e. within scripts.

Additionally, library functions for error handling and freeing memory are provided.

All functions with their parameters and return values are explained in detail in the User Manual to this software. The commandline applications are covered there as well.

2.2 Input

Only input in plain text format can be processed. The text documents for training (learning) have to be encoded in UTF-8, for classification this condition is not required but recommended.

The input can be passed to **Textweiser** either as a string or (a path to) a file.

The input's length is only limited by the respective data types.

Version 1.1.0 supports classification of the following languages:

→ German

→ English

Additional languages can be added upon request.

Unsupported languages can be processed as well, but results are likely to be less precise.

2.3 Return Value and Results

The return value of the main functions is an error indicator.

The results (of functions that have a result) are stored in a user-defined memory location that is passed to the function as an argument.

The functions for classification provide an array of data structures containing categories along with their probabilities, ordered descending by probability.

The data structure is formally defined as follows:

```
typedef struct {
    char    * category;    /* category name */
    float   probability;  /* probability */
} tw_prob_t;
```

2.4 Error Handling

Textweiser provides thread-safe error handling facilities. Each thread has its own error indicator that stores numeric error codes. Given these error codes, an English error message may be generated by a library function. For convenience, a named constant is defined for each error code, so that error names can be used instead of codes as well.

2.5 Security Considerations

Textweiser was designed and implemented with security in mind and has no single known safety defect. Besides that, the application does not evaluate any environment variables. Memory allocated internally by **Textweiser** is freed in all known error paths.

Textweiser supports encrypted connections to a database server if one of the following databases is used:

→ Microsoft SQL Server 2008

2.6 Thread Safety

Textweiser is thread-safe: more than one thread may call the library's functions at the same time. Each thread takes care of its own error handling and allocates two variables on the *Thread-Local Storage* (TLS).

The classification can moreover be done simultaneously if the database software supports it (*SQLite*).

3 Restrictions

- At the state of the art, a classification of text documents cannot be guaranteed to be accurate in any case without restrictions. There may be documents that are classified incorrectly.
- **Textweiser**'s accuracy is decisively dependent on the training of the categories.
- Only supported languages can be classified accurately. Unsupported languages will be processed as well but are likely to have less precise results.
- Input can only be processed if it is available in plain text or has been preprocessed to this format before. Any such preprocessing is not part of **Textweiser**.
- **Textweiser** itself does not implement encryption to securely connect to a database server, but solely relies on the encryption capabilities provided for this purpose by the database and its driver.
- The software itself as well as the man pages are provided in English only.
- The manual is provided in both an English and German version, but as a PDF document only.
- There is no general guarantee for downward compatibility to previous versions.

4 References

- User Manual for **Textweiser** version 1.1.0
- Lingua-Systems' **Textweiser** product website,
<http://www.lingua-systems.com/text-classifier/textweiser-library/>
- The Unicode Standard,
<http://www.unicode.org/>
- RFC 2279: "UTF-8, a transformation format of ISO 10646",
<http://www.ietf.org/rfc/rfc2279.txt>
- SQLite,
<http://www.sqlite.org/>
- Microsoft SQL Server,
<http://www.microsoft.com/sqlserver/>

5 License Informationen

Copyright (c) 2010-2011 Lingua-Systems Software GmbH

This software is covered by a commercial license. It may be used under the terms you agreed upon with Lingua-Systems Software GmbH. Please refer to your Agreement for details.

Textweiser contains software developed by Dr. Martin Porter, which is licensed under the BSD license:

Copyright (c) 2001, Dr. Martin Porter
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of Dr. Martin Porter nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.