

# TimblServer: Tilburg Memory-Based Learner Server

**version 1.0**

Manual

*ILK Technical Report – ILK-10-02*

Ko van der Sloot  
Induction of Linguistic Knowledge Research Group  
Tilburg centre for Creative Computing  
Tilburg University

URL: <http://ilk.uvt.nl><sup>1</sup>

May 31, 2010

<sup>1</sup>This document is available from <http://ilk.uvt.nl/downloads/pub/papers/ilk.1002.pdf>. All rights reserved Induction of Linguistic Knowledge, Tilburg University and CLiPS, University of Antwerp.

# Contents

<b>1</b>	<b>GNU General Public License</b>	<b>1</b>
<b>2</b>	<b>Installation</b>	<b>2</b>
<b>3</b>	<b>Changes</b>	<b>4</b>
3.1	version 1.0 . . . . .	4
<b>4</b>	<b>Quick-start Tutorial</b>	<b>5</b>
<b>5</b>	<b>Software usage and options</b>	<b>6</b>
5.1	Command line options . . . . .	6
5.1.1	Server options . . . . .	6
5.1.2	Configuration file . . . . .	6
5.2	Server interface . . . . .	7

# Preface

This reference guide describes the server interface to TiMBL, the Tilburg Memory-Based Learner software package. To be able to use TimblServer, a working installation of TiMBL (version 6.3 or upward) should be present on your system. For information on the installation and use of TiMBL, see the TiMBL Reference Guide (Daelemans et al., 2010) and the TiMBL webpage<sup>1</sup>.

This is the first release of the separate TimblServer package, which includes some new functionality compared to TiMBL 6.2 and earlier versions. Although this release has been tested for some time in our research groups, the software may still contain bugs and inconsistencies in some places. We would appreciate it if you would send bug reports, ideas about enhancements of the software and the manual, and any other comments you might have, to `Timbl@uvt.nl`.

This reference guide is structured as follows. In Chapter 1 you can find the terms of the license according to which you are allowed to use TimblServer. The subsequent chapter gives some instructions on how to install the TimblServer package on your computer. Chapter 3 lists the changes that have taken place up to the current version. Next, Chapter 4 offers a quick-start tutorial for readers who want to get to work with TimblServer right away. The tutorial describes, step-by-step, a case study with a sample data set (included with the TiMBL software) representing the linguistic domain of predicting the diminutive inflection of Dutch nouns.

---

<sup>1</sup><http://ilk.uvt.nl/timbl>

# Chapter 1

## GNU General Public License

TimblServer is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 3 of the License, or (at your option) any later version.

TimblServer is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with TiMBL. If not, see <<http://www.gnu.org/licenses/>>.

In publication of research that makes use of TimblServer 1.0, a citation should be given of: *“Walter Daelemans, Jakub Zavrel, Ko van der Sloot, and Antal van den Bosch (2010). TiMBL: Tilburg Memory Based Learner, version 6.3, Reference Guide. ILK Technical Report ILK-1001 Available from <http://ilk.uvt.nl/downloads/pub/papers/ilk1001.pdf>”*

## Chapter 2

# Installation

The TimblServer package can be obtained as a gzipped tar archive from:

```
http://ilk.uvt.nl/timbl
```

Following the links from that page, you can download the file `timblserver-1.0.tar.gz`. This file contains the complete source code (C++) for the TimblServer program, the license, and documentation. The installation should be relatively straightforward on most UNIX systems.

Before installing TimblServer, please make sure that a recent Timbl version ( $\geq 6.3$ ) is installed on your system. (If in doubt, check the output of the command `Timbl -V`).

To install the package on your computer, unzip the downloaded file (> is the command line prompt):

```
> tar xzf timblserver-1.0.tar.gz
```

This will make a directory `timblserver-1.0` under your current directory.

Alternatively you can do:

```
> gunzip timblserver-1.0.tar.gz
```

and unpack the tar archive:

```
> tar xf timblserver-1.0.tar
```

Go to the `timblserver-1.0` directory, and configure the package by typing

```
> cd timblserver-1.0
> ./configure --prefix=<location.to.install>
```

If you do not use the `--prefix` option, TimblServer will try to install itself in the directory `/usr/local/`. If you do not have root access you can specify a different installation location such as `$HOME/install`

It is not obligatory to install TimblServer, but if you plan to install TiMBL-based extensions such as Mbt<sup>1</sup> or Tadpole<sup>2</sup>, installing TimblServer is recommended.

After configure you can build TimblServer

```
> make
```

---

<sup>1</sup><http://ilk.uvt.nl/mbt>

<sup>2</sup><http://ilk.uvt.nl/tadpole>

and (as recommended) install:

```
> make install
```

If the process was completed successfully, you should now have executable files named `TimblServer` and `TimblClient` in the directory `<location_to_install>/bin`, and a static library `libTimblServer.a` in the directory `<location_to_install>/lib`.

Within the `<location_to_install>` directory a subdirectory is also created: `share/doc/timblserver` where the `TimblServer 1.0` documentation can be found.

`TimblServer` should now be ready for use.

The e-mail address for problems with the installation, bug reports, comments and questions is `Timbl@uvt.nl`.

# Chapter 3

## Changes

### 3.1 version 1.0

This is the initial release of a separate TimblServer. Compared with the version included in Timbl (up to version 6.2) the following is changed:

- Some bugs are fixed (`-C <n>` didn't work as expected)
- a `--config` option is added to read a configuration file. This file can be used to specify a whole range of separated and unrelated TiMBL experiments all to run on the same TCP port.

## Chapter 4

# Quick-start Tutorial

The use of TimblServer subsumes a familiarity with TiMBL. The TimblServer commandline interface is the same as that of Timbl with a few additions, exemplified below.

If you want to run the examples and demos from this manual, you should act as follows. We will use the same examples as TiMBL, so copy all the files from `<location_to_install>/share/doc/timbl/examples` to some working location. (By default, TiMBL writes its results to the directory where it finds the data.) From the directory you copied the files to, issue the following command:

```
> TimblServer -f dimin.train -S 7000
```

This will start a TimblServer on port 7000 (if that port is occupied you might need to try another number).

The server can be used (from the same machine) over Telnet like this:

```
> telnet localhost 7000
```

The server responds:

```
Welcome to the Timbl server.
```

Try to classify an instance:

```
c =, =, =, =, =, =, =, =, =, +, p, e, =, ?
```

The server should respond with:

```
CATEGORY {T}
```

# Chapter 5

## Software usage and options

### 5.1 Command line options

The user interacts with TimblServer through the use of command line arguments. When you have installed TimblServer successfully, and you type TimblServer at the command line without any further arguments, it will print an overview of the most basic command line options.

```
TimBL Server 1.0.0 (c) ILK 1998 - 2010.  
Tilburg Memory Based Learner  
Induction of Linguistic Knowledge Research Group, Tilburg University  
CLiPS Computational Linguistics Group, University of Antwerp  
Tue Mar 16 14:08:22 2010
```

```
usage: TimblServer --config=config-file  
or TimblServer -f data-file {-S socket} {-C num}  
or see: TimblServer -h  
for more options
```

#### 5.1.1 Server options

`--config <file>`: Start a server with a range of TiMBL experiments as specified in the configfile.

This is the **preferred** way to use TimblServer. See Section 5.1.2. for details about the configfile.

`-S <portnumber>`: Starts a TiMBL server listening on the specified port number of the localhost. This option is only available to be backward compatible with Timbl 6.2 and earlier versions; the `--config` variant is preferred.

`-C <number>`: limit the number of parallel connections to `<number>`. The default is 25.

#### 5.1.2 Configuration file

The configfile contains lines of the general format `attribute=value`. Valid attributes are: `port` to set the TCP port and `maxconn` to set the numer of parallel connections of the server. The attribute `protocol` is reserved for future usage. (We plan to implement a HTTP interface.)

Besides attribute–value pairs, the configfile can contain any number of lines describing TiMBL experiments to be served. The lines take the form of `basename=' 'timbl training options' '`. `basename` is a name you can choose to identify the experiment. The `basename` is needed when you call the server to classify an instance.

`' 'timbl training options' '` is a string of options which are used to train Timbl. For example:

```
port=7000
maxconn=10
dimin0="-aIB1 +vdi+db+n -f dimin.train"
dimin1="-aIGTREE +vdi+db +D -f dimin.train"
dimin2="-aTRIBL +vdi+db+n -q3 -f dimin.train"
```

If you start `TimblServer` with this configfile, it will start on port 7000, with at most 10 parallel sessions. The server will train 3 different Timbl instances, identified by the names `dimin0`, `dimin1` and `dimin2`.

## 5.2 Server interface

When the `TimblServer` is running, it is waiting for input on the specified port number. When a client connects on this port number, the server starts a separate thread, processing any given commands, such as to classify a new example. A sample client program is included in the distribution. The client must communicate with the server using the protocol described below. After accepting the connection, the server first sends a welcome message to the client:

```
Welcome to the Timbl server.
```

After this, the server waits for client-side requests. The client can now issue five types of commands: `base`, `classify`, `set` (options), `query` (status), and `exit`. The type of command is specified by the the first string of the request line, which can be abbreviated to any prefix of the command, up to one letter (i.e. `b`, `c`, `s`, `q`, `e`). The command is followed by whitespace and the remainder of the command as described below.

```
base basename
```

tell the server to which base the next command(s) refer. The `basename` must be one of the `basenames` specified in the configfile. If no configfile was specified, i.e. `TimblServer` is started with the old-school `-S` option, then the `basename` is “default”. It is not necessary to specify this default name when testing (therefore preserving backward compatability with older Timbl versions).

```
classify testcase
```

`testcase` is a pattern of features (must have the same number of features as the training set) followed by a category string. E.g.: `small, long, 1, ??`.

Depending on the current settings of the server, it will either return the answer

```
ERROR { explanation }
```

if something has gone wrong, or the answer

```
CATEGORY {category} DISTRIBUTION { category 1 } DISTANCE { 1.000000 } NEIGHBORS
ENDNEIGHBORS
```

where the presence of the DISTRIBUTION, DISTANCE and NEIGHBORS parts depends upon the current verbosity setting. Note that if the last string on the answer line is NEIGHBORS, the server will proceed to return lines of nearest neighbor information until it prints the keyword ENDNEIGHBORS.

#### set option

where option is specified as a string of commandline options (described in detail in Chapter 5.1 below). Only the following commandline options are valid in this context: k, m, d, B, L, Q, w, v, x. The setting of an option in this client does not affect the behavior of the server towards other clients. The server replies either with OK or with ERROR {explanation}.

#### query

queries the server for a list of current settings. Returns a number of lines with status information, starting with a line that says STATUS, and ending with a line that says ENDSTATUS. For example:

```
STATUS
LENGTH          : 0
MAXBESTS        : 500
NULL_VALUE      :
TREE_ORDER      : G/V
DECAY           : Z
INPUTFORMAT     : Column
SEED            : -1
DECAYPARAM      : 1.000000
SAMPLE_WEIGHTS  : -
IGNORE_SAMPLES  : +
PROBALISTIC     : -
VERBOSITY       : F
EXACT_MATCH     : -
USE_INVERTED    : -
GLOBAL_METRIC   : Overlap
METRICS         :
NEIGHBORS       : 1
PROGRESS        : 100000
TRIBL_OFFSET    : 0
IB2_OFFSET      : 0
WEIGHTING       : GRW
ENDSTATUS
```

#### exit

closes the connection between this client and the server.

## References

Daelemans, W., J. Zavrel, K. Van der Sloot, and A. Van den Bosch. 2010. TiMBL: Tilburg memory based learner, version 6.3, reference guide. Technical Report ILK 10-01, ILK Research Group, Tilburg University.