

# TextSTAT 3.0 User Manual

Dear user, thank you for your interest in TextSTAT. TextSTAT is a tool for corpus linguistics to make and edit corpora, and create concordance lists and frequency lists. The following manual will guide you through the process of installing, running and getting started using TextSTAT step by step.

## Installing TextSTAT

Start by downloading the installer for your operating system from the official TextSTAT website at: <https://neon.niederlandistik.fu-berlin.de/textstat/> We provide prepackaged versions for MacOS (for Mac Silicon/ARM and Intel Macs), a version for Windows (10+) and a Debian package (x86). If necessary it is also possible to compile and build TextSTAT by yourself by downloading the source code from the TextSTAT website.

### Windows

**TextSTAT** for Windows is packaged as a standalone and portable executable. Simply unpack the downloaded .zip file, you will get the TextSTAT.exe application which you can place anywhere you want. It is recommended to move the file somewhere outside of the Downloads directory, for instance your Desktop, so you can find it more easily later. Double-click the TextSTAT application to open.

### MacOS

**TextSTAT** for Mac is packaged as a DiskImage installer. Make sure you have the right package for your machine: Apple Silicon (with M1 chip or later) or Intel. After downloading double click the TextSTAT.dmg file, a window will open with the Install TextSTAT disk. Drag the TextSTAT application to the Applications folder as indicated in the window.

**Important:** Mac OS has a built in Gatekeeper that stops your computer from executing software that has not been certified by Apple. Since we are a small project, this is not possible for us. However, it is possible to circumvent the gatekeeper. To avoid blocking please:

- open TextSTAT.app (probably you will be notified by Mac OS, that opening is not possible)
- Go to your System Settings > Privacy and Security > Security Settings (at the very bottom)
- In the settings window it will state that TextSTAT.app was blocked either because it is not from the App Store or because the developer could not be identified
- Click “Open Anyway”

You only need to do this once (only the first time you want to run TextSTAT). More information on this can be found on Apple's support pages: <https://support.apple.com/guide/mac-help/open-a-mac-app-from-an-unidentified-developer-mh40616/mac>

### Linux (Debian, Ubuntu)

After downloading install the debian package from the terminal by running `sudo dpkg -i ~/Downloads/TextSTAT.deb`.

## Getting started: Overview of TextSTAT

After opening TextSTAT for the first time we recommend you to set the language first, which you can do in the language menu in the top menu bar.

### Buttons in the main window

The most important features are conveniently available through these buttons:

-  This button creates a new empty corpus.
-  This button opens an existing TextSTAT 3 corpus.
-  This button saves the current corpus.
-  This button adds an existing TextSTAT 3 corpus to the current corpus.
-  This button opens the TextSTAT web crawler to add websites to the current corpus.
-  This button adds a local file (.htm(l), .odt, .doc., .pdf, .txt, .md) to the current corpus.
-  This button removes the file selected in the corpus tab from the current corpus.
-  This button opens the word frequency tab and generates a default frequency list.
-  This button copies the selected item to the clipboard.
-  This button displays information about this app.

### Displaying your corpus data

There are four tabs that display different aspects of your corpus data:

1. Corpus
2. Word Forms
3. Concordance

## 4. Citation

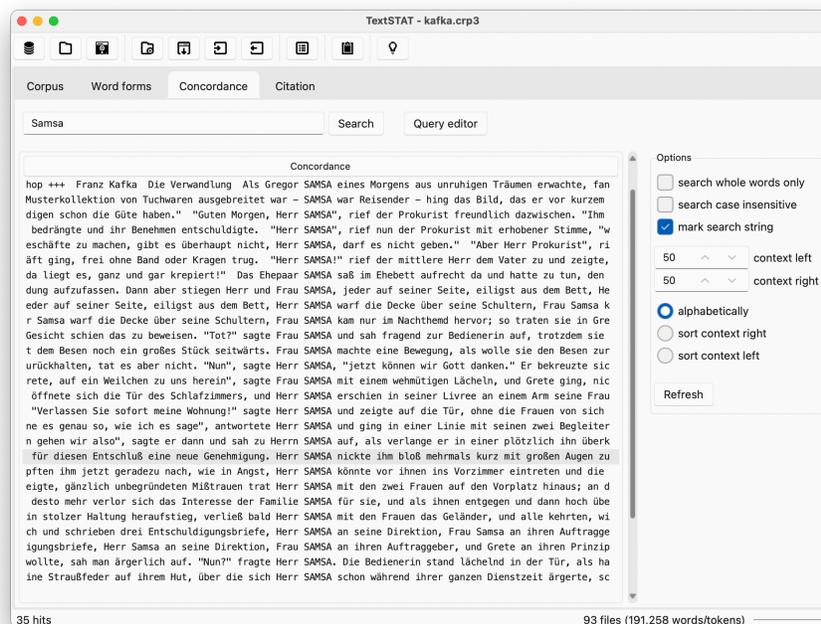


Figure 1: The main window of TextSTAT

**Corpus** In this tab you will find the list of files in your corpus. You can add plain text files, Word and OpenOffice files, and PDFs from your computer. You can also add HTML files directly from the web. Open the web dialogue (by clicking ) and insert an URL. After adding corpus files they will be converted to plain text and added tot the database. You can edit and delete corpus files by right-clicking them in the corpus tab.

**Word Forms** In this tab there is a table with a list of word forms in your corpus together with their respective frequencies. Clicking  will generate a list of all word forms. This list can be filtered and sorted. Double-click a form in the list to see all occurences as KWICs (Key Word In Context). To export the frequency list, use the Export menu in the top menu bar.

**Concordance** The Concordance tab displays a table with a list of keywords in context (KWIC). Search your corpus with the search bar on top. The search function supports regular expressions. The resulting list can be filtered and

sorted, context of the keywords can be extended or reduced. To export the concordance list, use the Export menu in the top menu bar. To see the key word with more context double-click the table-row in the list.

**Citation** The citation tab displays a KWIC within extended context. It also allows you to open the file of origin.

### Using Regular Expressions

When searching for certain patterns in the Concordance tab, you can use so-called 'regular expressions'. While these are not particularly user friendly, they are extremely powerful in executing very precise search queries.

#### Important special characters used in regular expressions

. (the dot) stands for any character you like \w stands for any alphanumeric character \W stands for any non-alphanumeric character (like space, punctuation marks) + the preceding character is repeated once or any number of times \* the preceding character is repeated any number of times, including zero \*? and +? make sure that \* and + are not 'greedy' (see examples) | stands for 'or' [] square brackets define a set of characters which are searched for alternatively {2,4} defines a range for the preceding character, e.g. at least 2 times but not more than 4 times

#### Examples

`b\wt` finds 'but', 'bit', 'bet' and 'bat' `b\w+t` finds 'but', 'bit', 'bet', 'bat', 'boat' and 'built' `w[ao]nder` finds 'wander' and 'wonder' `(this|that)` finds 'this' or 'that'

In the text *My sister sold me her house*

- `sol.+e` finds the string 'sold me her house'
- `sol.+?e` finds the string 'sold me'
- `s.+r` finds the string 'sister sold me her'
- `s\w+r` finds the string 'sister'
- `swe{1,3}t` finds 'swet', 'sweet' and 'sweet' (but not 'sweeet')

#### Tutorials

This only touches the surface of what you can do with regular expressions. However, there are several useful tutorials available online. We recommend e.g. [RegexOne](#) or [RegExr](#) or [regex101](#).