

Phase

3


PROJECT NN

Dokumentation der Software

PROJECT NN

Vokalerkennung mit Hilfe von neuronalen Netzen

Telefon 00 00/00 00 00 • Fax 00 00/00 00 00



Allgemeine Programmstruktur

Die Hauptaufgabe der ersten Entwicklungsphase bestand darin, die einzelnen Programmschritte sinnvoll in Klassen zu gliedern und deren Kommunikation untereinander zu bestimmen. Die spätere Softwareentwicklung wurde später anhand dieser erarbeitenden Struktur durchgeführt.

Eine Grafik über die allgemeine Struktur ist in Bild 1 zu erkennen.

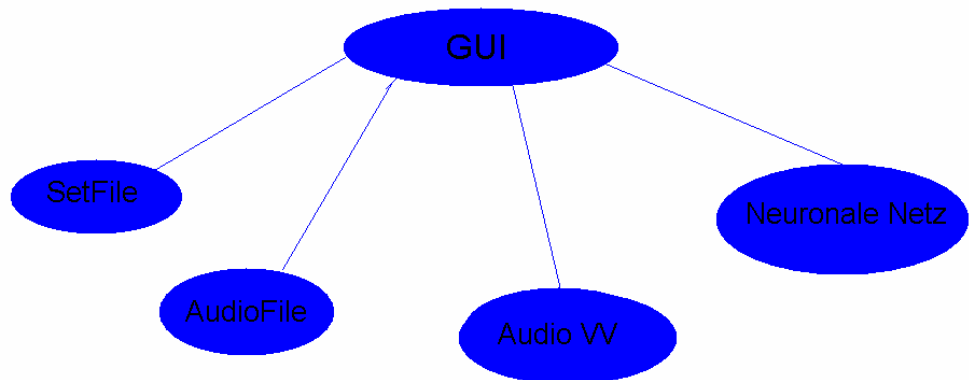


Bild 1: Klassenhierarchie

GUI

Jegliche Kommunikation zwischen den Klassen wird über die GUI durchgeführt. Sie regelt den Ablauf, liefert die zu verwendeten Dateien und benutzt die vom Neuronalen Netz erhaltenen Statistikdaten zur Visualisierung.

Im Laufe der Entwicklungsarbeit wurde es jedoch notwendig das Programm auch ohne GUI zu testen. Dann wurde die Kommunikation von einem separatem Programm (z.B. main()) geregelt.

SetFile

Um eine Liste von Audiodateien zu verwalten und zu organisieren, musste eine Dateiliste mit Abspeicherung in einer Datei ermöglicht werden. Die SetFile Klasse regelt die gesamte Organisation dafür.

AudioFile

Die zu verwendeten .au Dateien unterliegen bestimmten Formatsvorlagen vor der Verwendung in unserer Software (z.B. muss die Samplerate 11025 sein). Diese Klasse prüft die einzuladenden Dateien und übergibt den Dateninhalt plus den im DateiInfoBlock abgespeicherten Vokal.

AudioVV

Die Klasse AudioVV übernimmt die Audiovorverarbeitung. Die von der AudioFile Klasse erhaltenen Daten werden dekodiert und für das Neuronale Netz aufbereitet.

NeuronaleNetz

Das Neuronale Netz bekommt die dekodierten Daten von AudioVV, verarbeitet diese mit den gegebenen Algorithmen und lernt und/oder testet so die Vokalerkennung.

Nähere Informationen zu den Klassen im Laufe der Dokumentation.

Benutzung der Software

In der MainGUI wird der erste Arbeitsschritt ausgewählt. Sie zählt als Tor zu den weiteren Unterschritten der Bearbeitung.

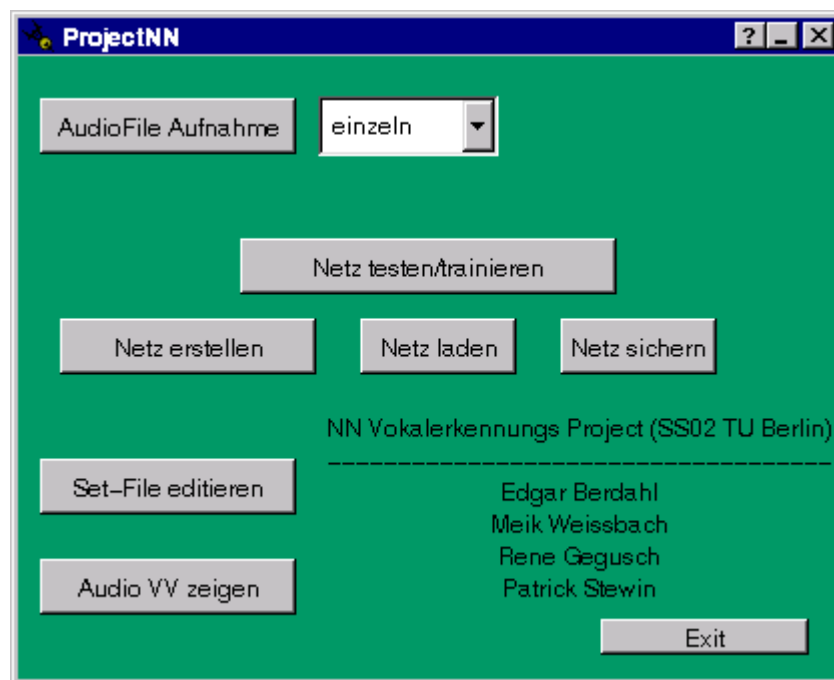


Bild 2 MainGUI

Zuerst sollte eine oder mehrere .au Dateien aufgenommen werden. Dazu wählt man den entsprechenden Button und in der ComboBox die Auswahl, ob nun mehrere Dateien aufgenommen werden sollen, oder nur eine einzelne Datei.

Eine einzelne Audiodatei aufnehmen.

AudioRecordGUI dient der Aufnahme einer einzelnen Audiodatei mit Hilfe von audiorecord. Außerdem kann man hier auch die gerade aufgenommenen Daten visualisieren lassen.

Hier muss man auch angeben, welcher Buchstabe aufgenommen werden soll, dies ist später wichtig für das Netz zum Lernen. Zusätzlich kann man die Länge und die Lautstärke der Aufnahme einstellen. Nachdem die Aufnahme beendet ist, kann man sie sich noch einmal anhören und dann bei Gefallen abspeichern.

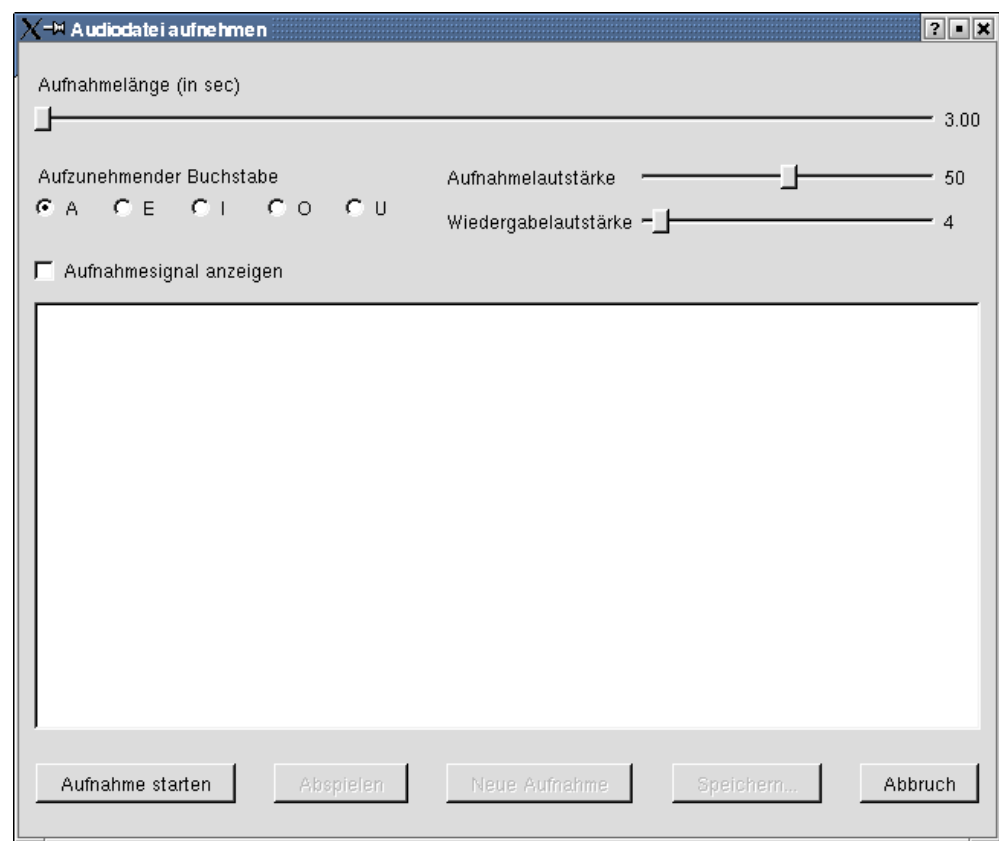


Bild 3 AudioRecordGUI

Mehrere Audiodateien aufnehmen.

SetFileRecGUI dient der Aufnahme mehrerer Audiodateien mit Hilfe von audiorecord. Auf eine Visualisierung der Audiodaten haben wir hier verzichtet.

Als erstes muss man eine Vorsilbe und ein Verzeichnis für alle .au-Dateien angeben. Desweiteren gibt man an, wieviele Dateien man aufnehmen möchte. Für die einzelnen Aufnahmen sind die Einstellungen die gleichen, wie in AudioRecordGUI.

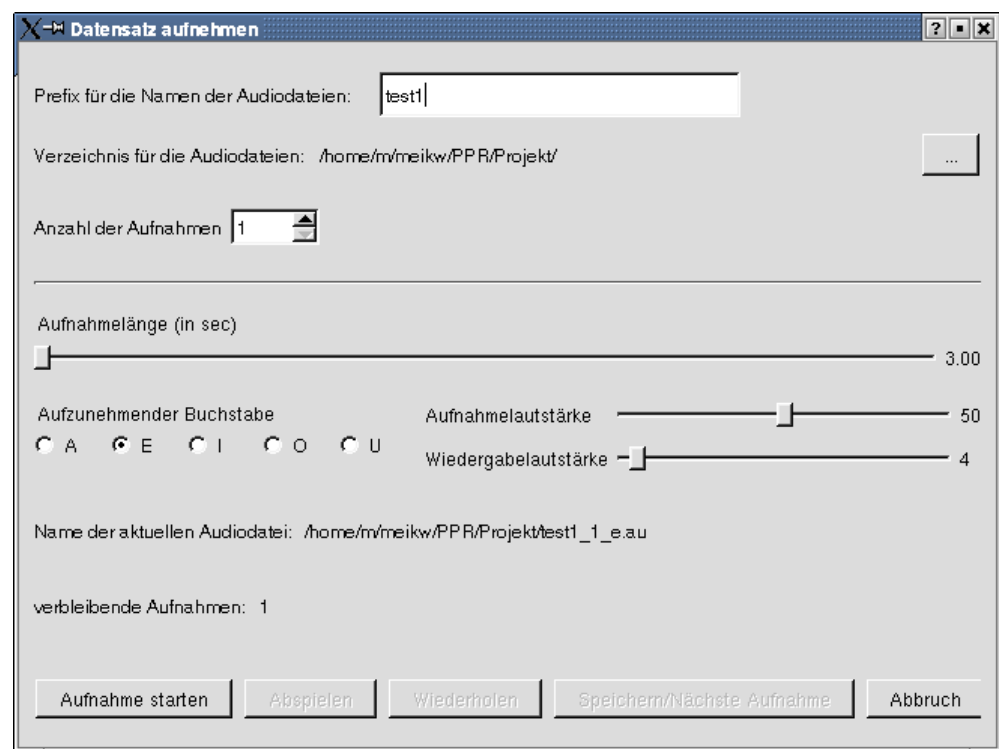


Bild 4 SetFileRecGUI

Setdateien erstellen und bearbeiten.

Möchte man später mehrere .au-Dateien als einen Datensatz einlesen, muß ein Set-File eingelesen werden.

SetFileEditGUI ermöglicht einerseits die Zusammenstellung von mehreren .au Dateien zu einem Set-File und andererseits ist es auch möglich, einen bestehenden Datensatz zu editieren.

Als erstes wählt man eine .set-Datei aus. Wenn man hier den Namen einer nicht existenten .set-Datei eingibt, legt man einen neuen Datensatz an. Wählt man eine bestehende .set-Datei aus, wird deren Inhalt (die Pfade der .au-Dateien) im linken Fenster angezeigt. Dann wählt man ein Verzeichnis mit .au-Dateien, die in diesem Verzeichnis enthaltenen .au-Dateien werden dann im rechten Fenster angezeigt. Man kann dann im rechten Fenster eine oder mehrere Dateien markieren und durch Klick auf die Pfeiltaste in der Mitte, die Namen dieser Dateien ins SetFile einfügen. Durch Klick auf die "Entf."-Taste in der Mitte kann man die im linken Fenster markierten Pfade aus der .set-Datei entfernen.

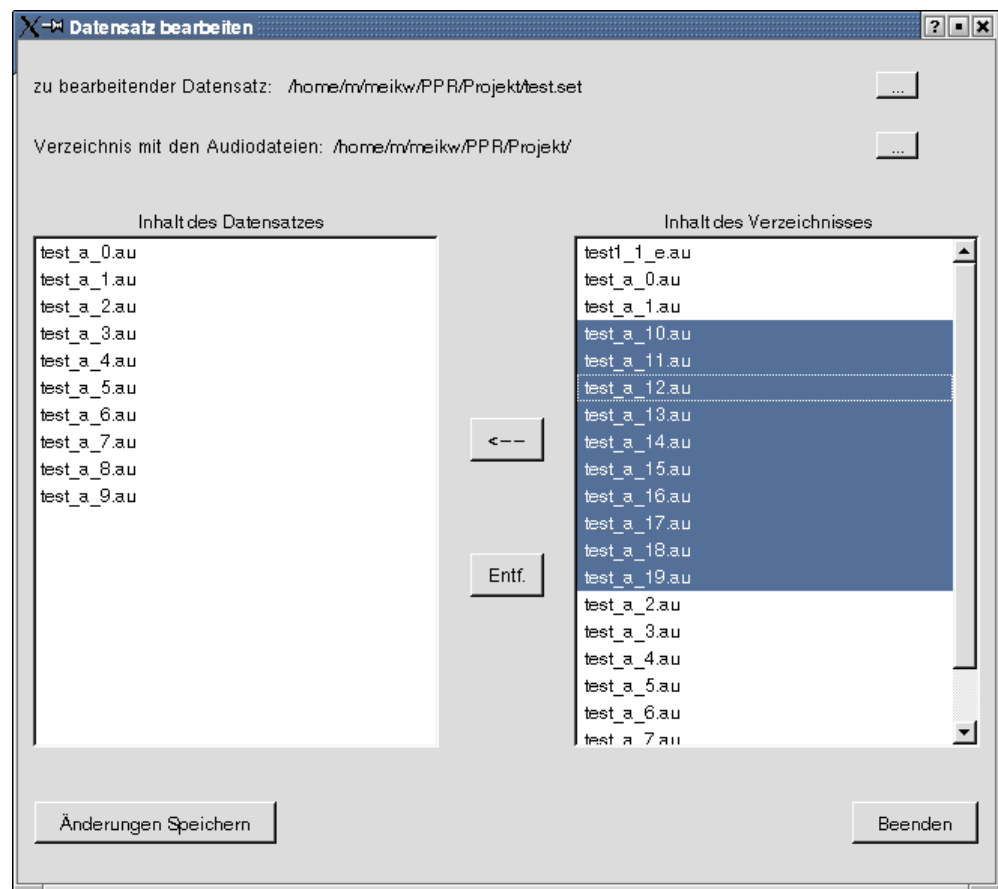


Bild 5 SetFileEditGUI

Audiovorverarbeitung visualisieren.

AuEditGUI dient der Visualisierung der Audiovorverarbeitung für eine Audiodatei.

Man wählt eine .au-Datei aus, stellt die gewünschten Parameter für die Audiovorverarbeitung ein und klickt dann auf "Weiter". Dann öffnet sich ein neues Fenster, das aus 3 Teilfenstern besteht. Im oberen Teilfenster sind die Originaldaten als 8192-Byte-Ausschnitt zu sehen, im mittleren Teil sind die Daten nach der FFT zu sehen und im unteren Fenster sind die Daten nach der Hamming-Fensterung zu sehen.

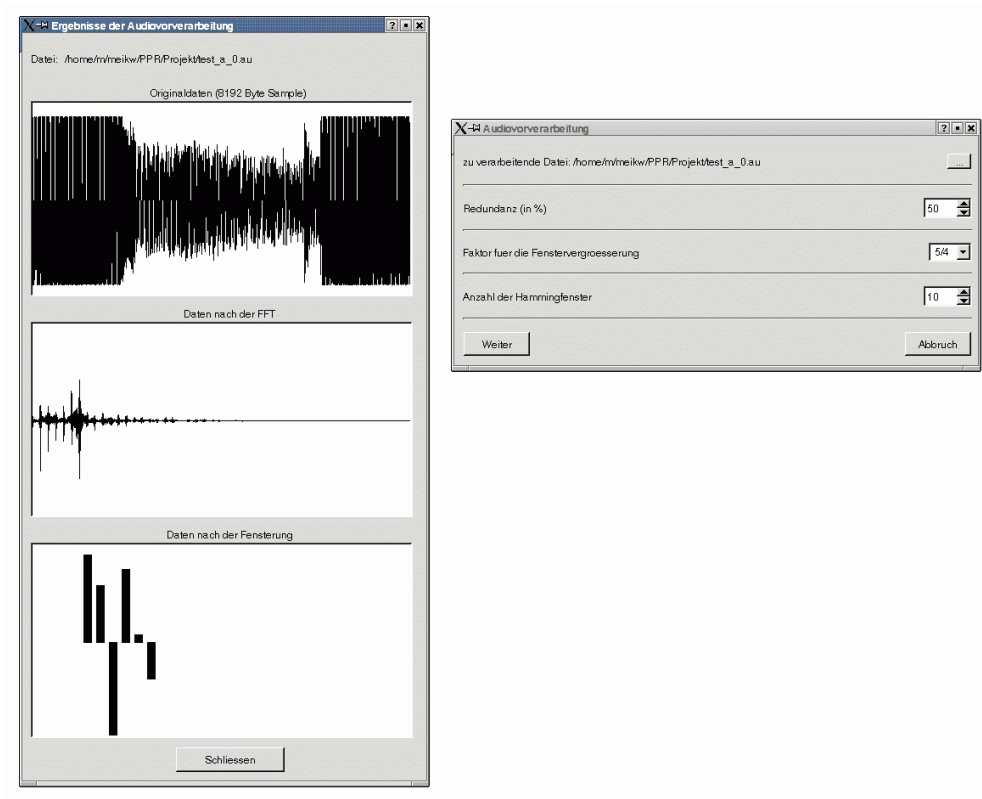


Bild 6 AuEditGUI

Ein neuronales Netz trainieren oder testen.

Diese GUI übernimmt im Endeffekt zwei Aufgaben und zwar das Testen und das Trainieren. Unser Entschluß war, in beiden Fällen dieselbe Ausgabe zu ermöglichen.

Zum Trainieren wird ein Trainingsdatensatz ausgewählt und die entsprechenden Anzahlen von Epochen, Lernschritten usw. ausgewählt. Über den START Knopf wird der Vorgang gestartet. Anschließend werden die Fehlerkurven, Gewichts- und Schwellmatrix sowie die Ergebnisstatistik anhand eines Balkendiagramms angezeigt.

Soll ein Netz trainiert werden, so sind folgende Schritte notwendig:

- Trainingsdatensatz auswählen (möglichst ein SET-File mit vielen Trainingsdateien, z.B. 250)
- Testdatensatz zum Testen auswählen (SET File mit z.B. 50 Dateien)
- Anzahl der Lernschritte und Epochen wählen
- Batch- oder Online Learning auswählen
- Und ein Netz wählen

Zum Testen, entsprechend die zu testende Datei auswählen, anstelle eines SET Files oder ein SET File auswählen (in diesem Fall werden die Ergebnisse für nur die letzte Datei im SET File angezeigt) und mit den unteren Netzparametern starten. Das Testergebnis wird in der GUI für Wahrscheinlichkeiten der Netze ausgegeben.

Online Learning heißt, dass die Werte im Netz nach der Untersuchung jeder Datei geupdatet werden. Beim Batch Learning wird nur nach jeder Epoche geupdatet. In diesem Modus läuft es ein bisschen schneller, aber das Netz lernt entsprechend langsamer, und es ist wahrscheinlicher, dass die Fehlerkurven dann ein leicht oszillieren.

Redundanz und Faktor für die Fenstervergrößerung sind Parameter der Audio Vorverarbeitung und ihre Wirkungen können in der Doku für die Audio Vorverarbeitung nachgeschlagen werden. Die Werte im Bild 7 funktionieren ziemlich gut.

Die nachfolgenden Bilder stammen aus einem Beispiel, in dem das Netz zuerst gut trainiert wurde (der Fehler war ungefähr 0.02) aber danach bei anderen nicht geeigneten Parametern trainiert wurde.

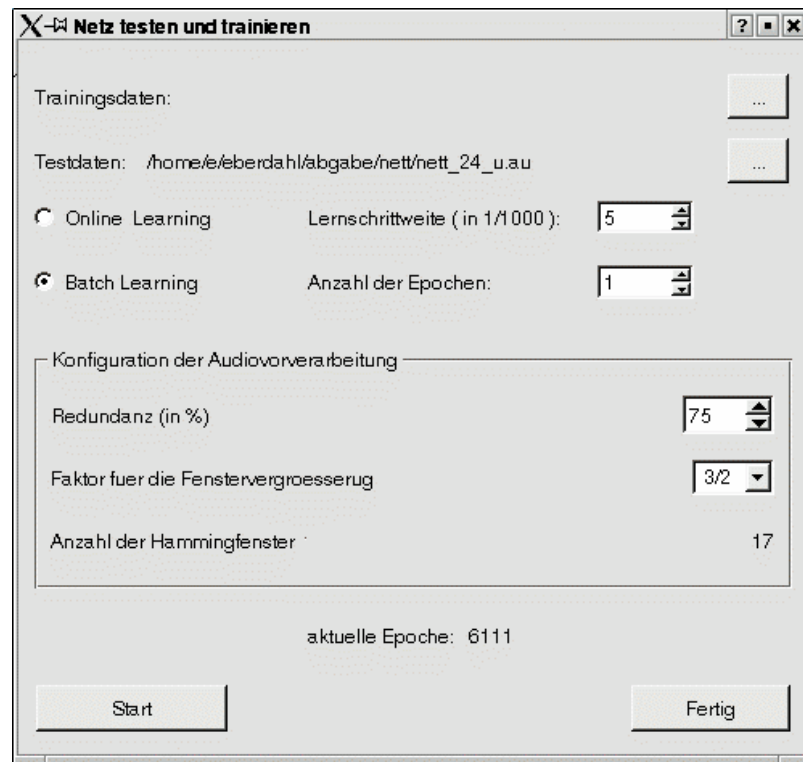


Bild 7 NetzTrainieren/testenGUI

Fehlerkurven

Die Fehlerkurven werden nach Auswahl eines Netzes über den Knopf AKTUALISIEREN angezeigt. Sind die Fehlerkurven für das Anzeigefenster zu lang, so werden sie in ihrer Länge gestaucht. Beide Kurven werden in der gleichen Länge angezeigt. Sollte eine der beiden Kurven länger sein, als die andere, so wird die längere nicht weiter gezeichnet als die andere lang ist.

Die rote Kurve stellt die Testfehler dar, und die blaue die Trainingsfehler. Die Testfehler sind natürlich meistens höher, da das Netz nur auf den Trainingsdatensatz trainiert wird. Die horizontale Achse entspricht der Zeit, und die vertikale Achse entspricht den durch

$$\text{Dim}(1)$$

$$E = (1/2) * \sum_{i=1} (y_i^T - y_i^L)^2$$

berechneten Fehlern. y_i^T ist der Soll-Output, y_i^L ist der Ist-Output des Netzes und 1 ist die Anzahl der Dimensionen der Ausgabe. Im Bild ist die horizontale Achse die Zeit und die vertikale Achse der Fehlerwert (zwischen 0.0 und 0.5).

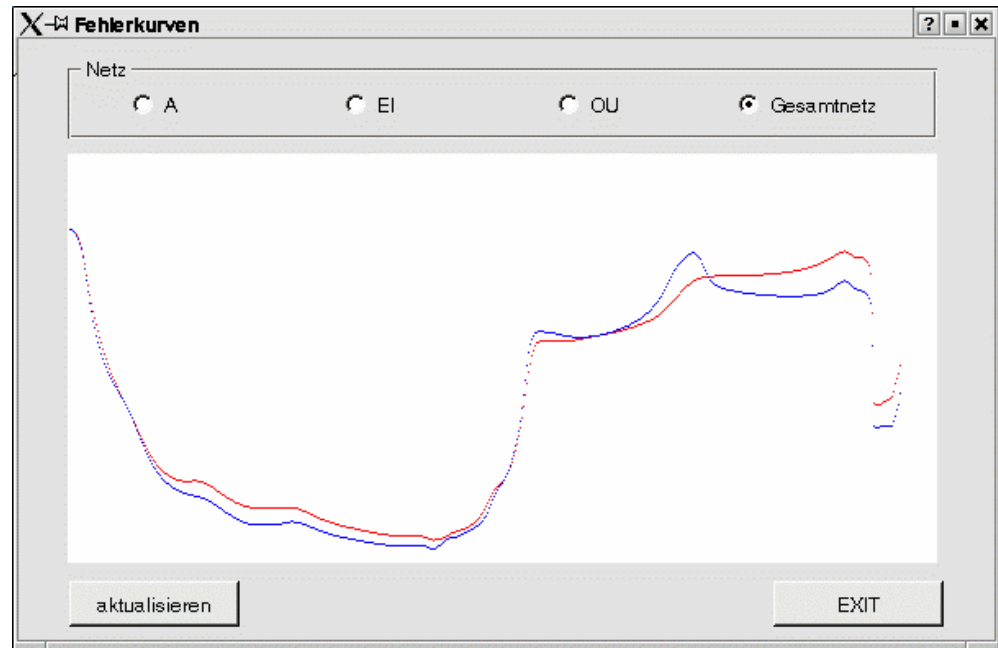


Bild 8 FehlerkurvenGUI

Gewichts- und Schwellmatrix

Zur Anzeige muß das Netz und die Schicht ausgewählt werden. Über den Knopf AKTUALISIEREN werden die Matrizen gezeichnet.

Sollte eine nicht zur Verfügung stehende Schicht ausgewählt werden, so wird die zuletzt gewählte Schicht verwendet!

Die Gewichtsmatrix steht auf der linken Seite und stellt die Werte der Gewichte in den Neuronen dar. Schwarz bzw. grün entsprechen -1.0 bzw. 1.0 . Die vertikale Achse vertritt die Eingabe zur Schicht, und die horizontale Achse die Ausgabe der Schicht. Rechts stehen die Schwellenwerte. Schwarz ist -1.0 und blau ist 1.0 .

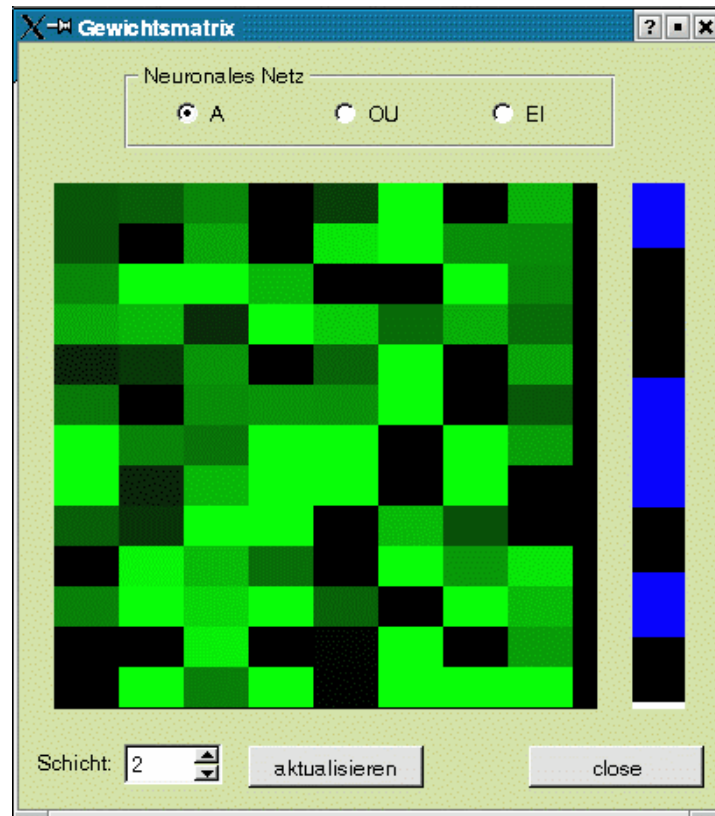


Bild 9 GewichtsmatrixGUI

Wahrscheinlichkeiten

Hier wird angezeigt, welcher Vokal mit welcher Pseudoahrscheinlichkeit für das gesamte und die einzelnen Teilnetze erkannt wurde.

Außerdem wird in einer Textzeile der erkannte Vokal mit seiner Pseudowahrscheinlichkeit angezeigt.

Es soll beachtet werden, dass diese Werte keine echten Wahrscheinlichkeiten im mathematischen Sinne sind. Sie sind einfach Ausgaben des Netzes, die auf skaliert werden, damit sie zur 1.0 aufaddieren.

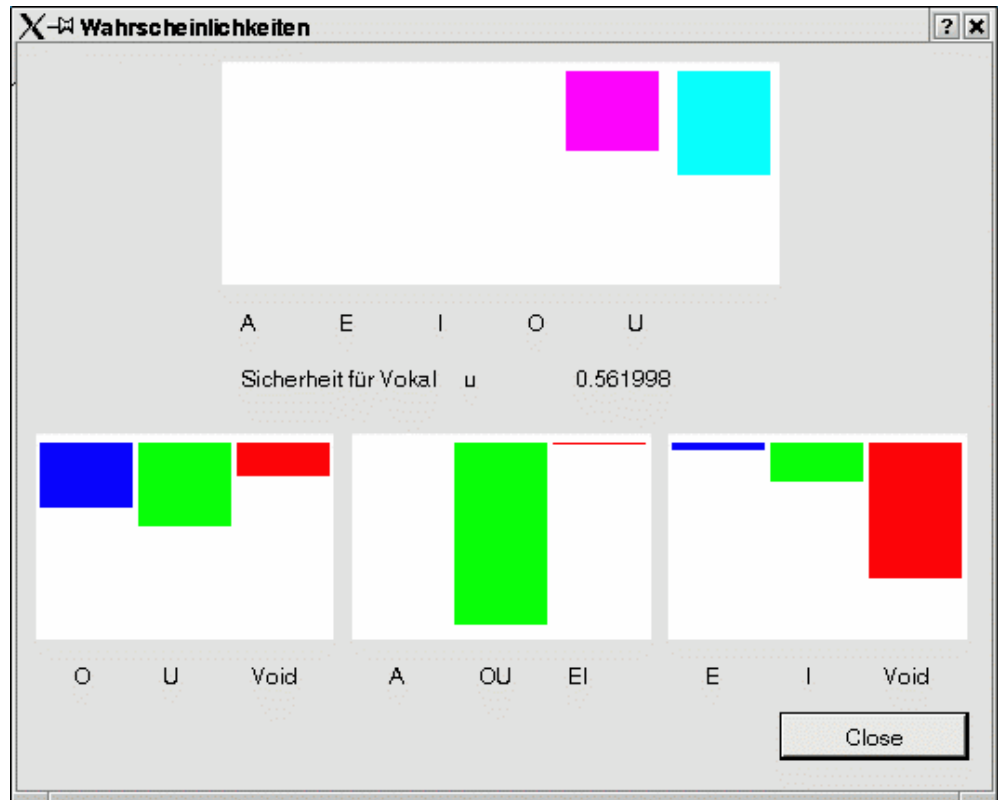


Bild 10 WahrscheinlichkeitenGUI

Hier ist ein Bild, welches zeigt, wie die Fenster aussehen, nachdem man trainiert hat.

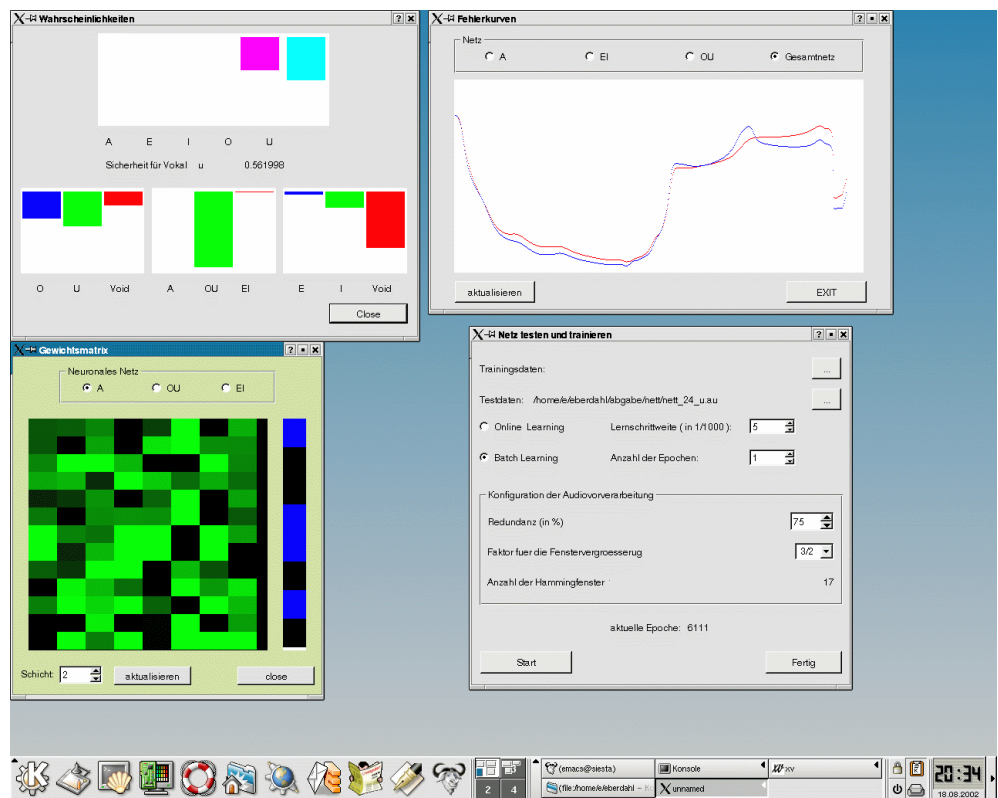


Bild 11 Alles Zusammen

Ein Netz neu erstellen

Ein neues Netz zu erstellen verlangt die Einstellung aller Netze und deren Schichten.

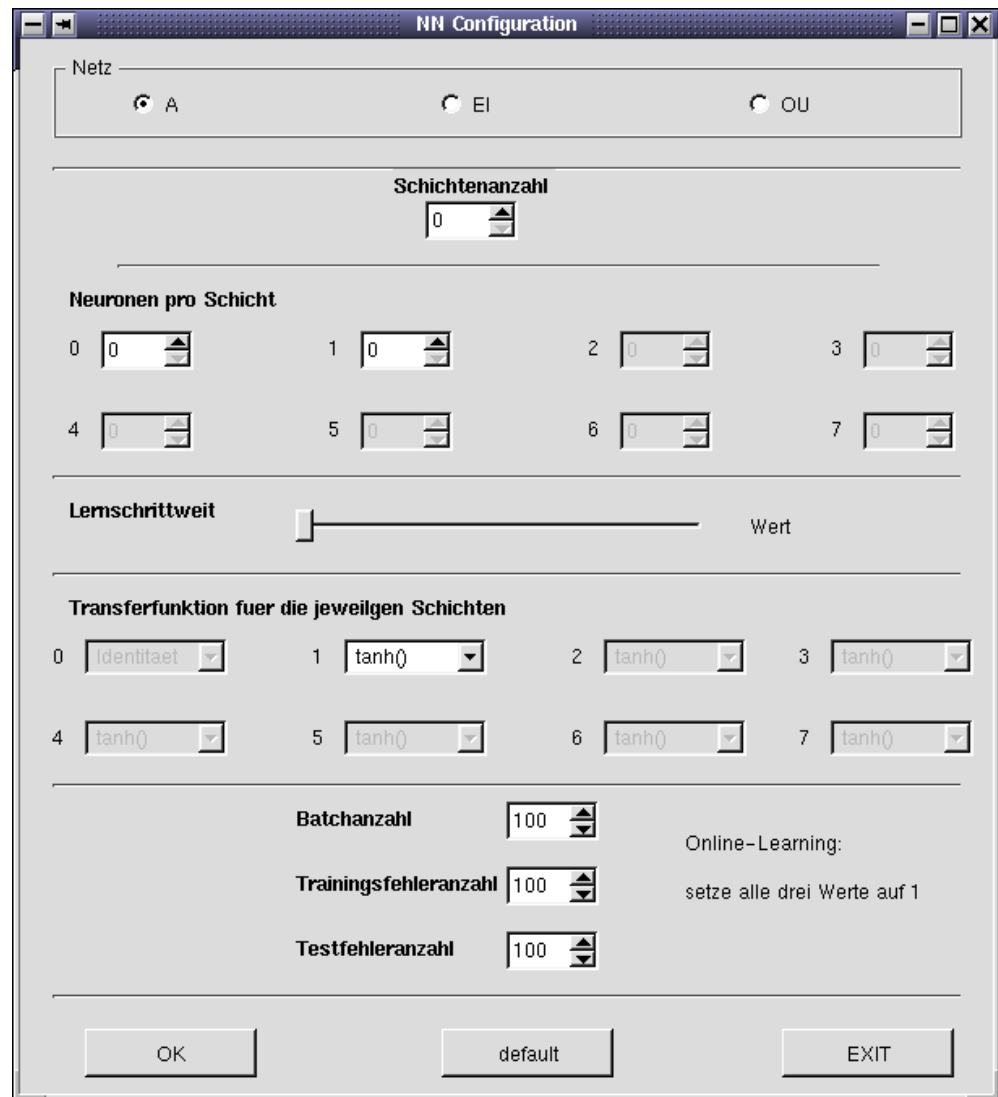


Bild 11 – NN erstellen

Das Bild zeigt schon recht eindeutig welche Parameter für die Netze zu machen sind. Es ist wichtig, dass alle Schichten für jedes Netz unabhängig eingestellt werden.

Hinweis:

Lernschrittweite, Batchanzahl, Trainingsfehler und Testfehleranzahl haben keine Funktion mehr. Sie sind im Laufe der Entwicklung deaktiviert worden