



Assistive Context-Aware Toolkit (ACAT)

Developer Guide

Version 2.00

Table of Contents

Overview.....	3
What is ACAT?	3
Key features	3
Architecture	3
ConvAssist	4
Building ACAT	5
System Requirements	5
Build Instructions	5
Running the application	6
Building the installer for ACAT.....	6
ConvAssist	7
Overview.....	7
Modes of Interaction	7
Sentence Mode.....	7
Canned Phrases Mode.....	8
Short-Hand Mode.....	8
Building ConvAssist on Windows.....	8
Install Python	8
Create the virtual environment	8
Build ConvAssist.exe	9
Deactivating the virtual environment.....	9
Building ConvAssist on Mac or Linux.....	10

Overview

General overview of functionality and application of this software

What is ACAT?

Basic features of ACAT

Assistive Context-Aware Toolkit (ACAT) is an open-source platform developed at Intel Labs to enable people with limited ability to speak and type, to communicate easily through keyboard simulation, word/sentence prediction and speech synthesis.

Key features

- [Switch Scanning](#) Interface for Augmentative and Alternative Communication (AAC)
- Range of input modalities to support varying user physiological constraints
- Predictive language model to reduce typing effort
- Extensible platform for engaging developer community

ACAT was originally developed by researchers in Intel Labs for Professor Stephen Hawking. Professor Hawking was instrumental to the design process and was a key contributor to the project design and validation. After Intel deployed the system to Professor Hawking, we turned our attention to the larger community and continued to make ACAT more configurable to support a larger set of users.

By open sourcing this configurable platform, we intend for developers to continue to expand its capabilities by adding new user interfaces, new sensing modalities, language prediction and other features.

Architecture

ACAT is structured as a plug-in framework. This enables developers to extend the capabilities by adding Extensions aka plug-ins. The extensions are dynamically discovered and loaded at runtime. They provide services such as text-to-speech conversion, handling input switches, word and sentence prediction, etc.

The high-level architecture is shown in Figure 1. The ACAT Core library and the ACAT Extension library provide all the core functionalities. They enumerate and load the various extensions at runtime. Extensions raise events to notify subscribers when something interesting happens.

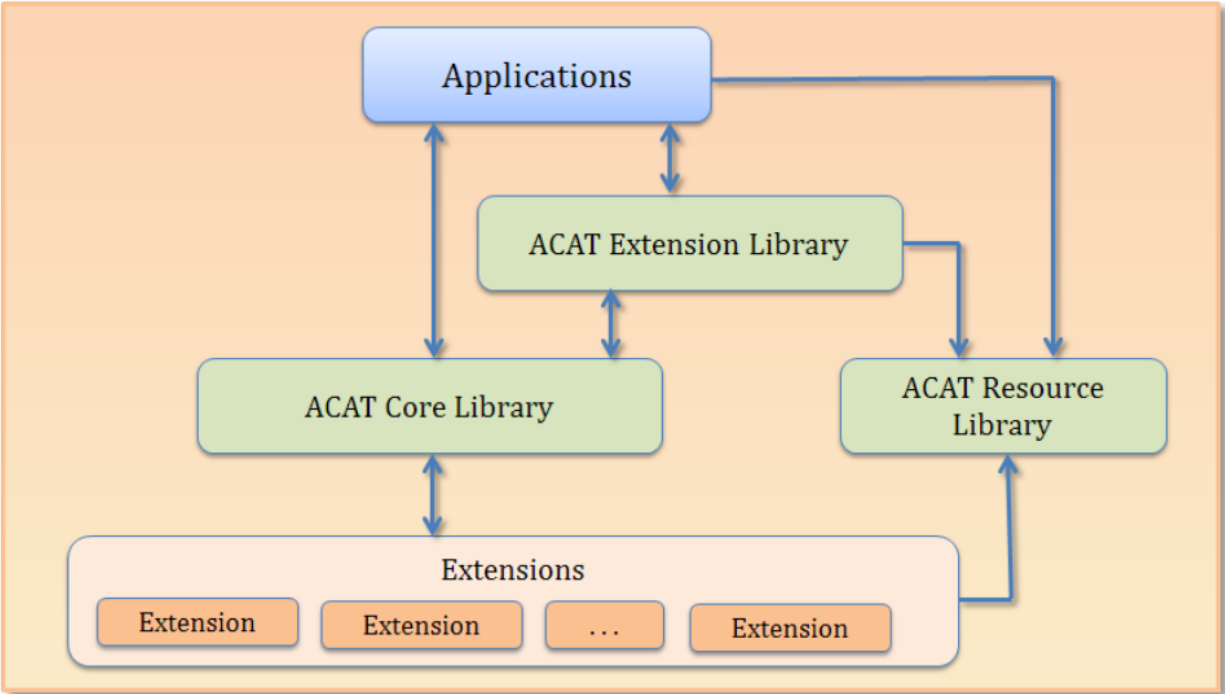


Figure 1: ACAT architecture

ConvAssist

The ConvAssist module, developed by Intel Labs, provides word and sentence predictions for ACAT. It extends Pressagio, which predicts text based on n-gram models. ConvAssist uses language models, n-gram databases, semantic match modules, and various safety filters that have been fine-tuned for Augmentative and Alternative Communication (AAC).

ConvAssist can also be tested as a stand-alone application. It is extensible and allows developers to bring in new predictors, new language models for word prediction, sentence completion, semantic search, and more. It currently supports English-only models but can also be extended to other languages by incorporating multilingual language models.

The remainder of this document contains build instructions for ACAT and ConvAssist.

Building ACAT

System Requirements and building the application

The ACAT application and its libraries have been developed in C# using WinForms for the user interface.

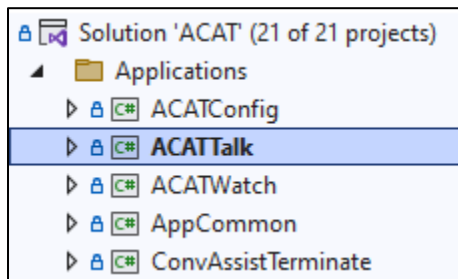
System Requirements

Requirements for building the solution are:

- Windows 10 20H2 or higher / Windows 11
- Visual Studio 2019 or higher
- At least 15 GB of free disk space
- At least 8 GB RAM
- .NET 4.8.1

Build Instructions

1. Clone the ACAT repository (master branch).
2. Navigate to the **src** folder.
3. Open **ACAT.sln** in Visual Studio.
4. Set **ACATTalk.exe** as the Startup project.



5. Select the build configuration (Release or Debug).
6. Do a **Rebuild All**.
7. This will build all the core libraries and extensions, and ACATTalk.exe which is the application.

Running the application

You can the ACAT Talk application from within Visual Studio. To manually run it:

1. Right click on the ACATTalk project and select **Open Folder in File Explorer**.
2. Navigate to **`$\src\Applications\ACATApp\bin\`(Config)** folder where Config is either Release or Debug depending on the solution configuration.
3. Run **ACATTalk.exe**.

ACAT stores all user related data such as settings, personalized phrases, personalized word prediction databases, etc. under the ACAT folder in the user's Documents folder.

Building the installer for ACAT

ACAT uses NSIS (Nullsoft Scriptable Install System) to create ACATSetup.exe, the installer.

You must install Python v3.12 or higher to create the installer.

Follow these instructions to create the ACAT installer.

1. Download NSIS from here: <https://nsis.sourceforge.io/Download>
2. Do a Release build of ACAT using the Build Instructions in this section.
3. Go to the Release folder (**`$\src\Applications\ACATApp\bin\Release`**), search for *.PDB files and delete them. They need not be included in the installer.
4. Go to the command prompt and navigate to the Setup folder in the ACAT source tree. This is under the **`$\src`** folder.
5. Run the following command:
`python installGenerator.py <path to the ACAT binaries>`
Here, path to the ACAT binaries is the folder that has the ACAT build which is under **`$\src\Applications\ACATApp\bin\Release`**
6. This will create the NSIS script file. The file is called **NSIS_ScriptInstaller.nsi**
7. Launch the NSIS compiler.
8. Open the **NSIS_ScriptInstaller.nsi** and compile it.
9. At the end of compilation, NSIS will create **ACATSetup.exe**

ConvAssist

General overview of this module and building it

Overview

ConvAssist is a module developed by Intel Labs. It enables users with disabilities to communicate using the latest language modeling technologies, through ACAT. ConvAssist can quickly suggest utterances/sentences that the user can use to have a social conversation or communicate with their caregiver in near real time.

ConvAssist is built on [Pressagio](#), that is a library that predicts text based on n-gram models. Pressagio is a pure Python port of the Presage library and is part of the [Poio](#) project.

ConvAssist runs as an independent process. The executable ConvAssist.exe can be found in the ConvAssistApp folder under `$(src)\Applications\ACATApp\bin\(\Config)` folder, where (Config) is either Debug or Release. The ACAT application launches ConvAssist on startup. After ConvAssist is launched, it runs in the systray. ACAT continuously provides ConvAssist the text that the user has entered in the ACAT Talk application window and ConvAssist returns the next word predictions and sentence completions, using the text entered so far as context. ACAT then displays the predicted words and sentence completions in the main window of the application. ACAT and ConvAssist communicate with each other through Named Pipes. Data is exchanged in the JSON format.

ConvAssist contains language models based on specific Assistive and Augmentative Communication (AAC) datasets, and dialog datasets, tailored for day-to-day communication in our assistive use case. These language models support both next word prediction and sentence completion for enabling near-real time communication with least amount of user effort and intervention.

Modes of Interaction

ConvAssist enables different modes of interactions for users through the ACAT interface. The three modes are:

1. Sentence
2. Canned Phrases
3. Short-Hand

Sentence Mode

In the Sentence Mode, users can use the interface to type any sentence they would like to. As they type a letter, word completions appear along with their probabilities.

After a word is chosen or typed, sentence completions also appear with 5 possible completions of the context typed so far. The predictions come from a hybrid system that does both database retrieval (from users personalized database and an AAC database) and generations from an AI model. The AI model is a GPT2 model finetuned for the AAC use case. Please find the associated details and the [model card here](#).

Canned Phrases Mode

The Canned Phrases Mode can be used to store and search for frequently used phrases for easy and quick access.

Short-Hand Mode

The Short-Hand Mode can be used by users for quick communication - in situations where the user needs to communicate with their caretaker with minimum number of words (without the need for grammatical sentences).

Building ConvAssist on Windows

The source code for ConvAssist can be found in the ACAT repository under src\ConvAssist.

Install Python

ConvAssist requires Python v3.11.5. Install it.

Create the virtual environment

From the command prompt, use this command to install the virtual environment tools

pip install virtualenv

Use the following command to create the virtual environment for a specific Python version (v3.11.5 in this case).

**virtualenv --python="PATH_TO_PYTHON_EXE_FROM_PROGRAM_FILES"
"PATH_TO_FOLDER_OF_ENV_MODULES"**

The first parameter is the location of Python.exe for the specific version. The second parameter is the location where the virtual environment files will be installed. Any module downloads will be in this folder and not in the main Program files of the user PC

For instance, for Python v3.11.5, here is the command.


```
virtualenv --python="\Path\to\pythonInstallation...\AppData\Local\Programs\Python\Python3115\python.exe" "C:\Python Projects\ ConvAssist Main\VirtualEnv 3115"
```

Use the following command to activate the virtual environment.

Navigate to the folder where the virtual environment files are located. This is the second parameter used to create the virtual environment (see above).

Go to the **Scripts** folder under that folder.

Type the following command:

activate

For example:

```
C:\Python Projects\ConvAssist Main\VirtualEnv 310\Scripts> activate
```

This will activate the virtual environment for the specific Python version. Once activated, any **pip** command to download a module will install the module in the environment.

Build ConvAssist.exe

1. Go to the ConvAssist source tree folder under ACAT. It is under **\$(src)\ConvAssist**.
2. Navigate to the **ACAT_ConvAssist_Interface** folder.
3. Use the command below to build ConvAssist.exe
pyinstaller ConvAssistCPApp.spec
This will build the ConvAssist.exe under the **ACAT_ConvAssist_Interface\dist** folder.

Deactivating the virtual environment

To deactivate the virtual environment, use the following command in the virtual environment command prompt.

deactivate

Building ConvAssist on Mac or Linux

ConvAssist works with Python 3.11.5. Create a virtual environment and install the required packages. You can then run ConvAssist as an independent application on Mac or Linux.

1. Clone the ACAT repository from GitHub.
2. Install Python 3.11.5
3. Install the virtual environment using this command
pip install virtualenv
4. Create a virtual environment using this command:
virtualenv myenv
5. Activate the virtual environment using this command
source myenv/bin/activate
6. In the ACAT source tree, navigate to **\$/src/ConvAssist**
7. Install requirements using this command
pip install -r requirements_py3115.txt
8. To use ConvAssist as a standalone application use this command
pip install wxPython
9. Go to the ConvAssist folder and type the following command
pip install -e .
10. Go to the **ACAT_ConvAssist_Interface** folder.
11. Type the following command to run ConvAssist
python pyprompter.py