Texas Christian University

ClassifAI Instructional Analysis Tool Glossary

Version 1.0

Revision History

Date	Version	Description	Author
06/05/24	1.0	Initial glossary release	Jaxon Hill

Table of Contents

Glossary

1. Introduction

This document serves as a glossary to any terms or terminology used in ClassifAI Documentation, which is not considered to be common knowledge for the layperson. This document will therefore define terminology specific to the problem domain and capture data definitions, to help keep other project documentation succinct and precise.

1.1 Purpose

To define terminology specific to the problem domain.

1.2 Scope

This document should handle the entire scope of the ClassifAI project, ranging from GitHub documentation, in-line comments in source code, all the way to official project documentation. Anything under the scope of the project may reference terms defined in this document.

1.3 References

huggingface.com - an open source forum for machine-learning related programming.

1.4 Overview

This glossary will not be strictly subcategorized, but every term will have a matching label next to it denoting the parent domain. Terms will fall under Education, Technology, or General. Terms will be in alphabetical order for ease of use.

2. Definitions

2.1 API [Technology]

Shorthand for Application Programming Interface. APIs define proper syntax for messages to and from software products, and at a more granular level, how different sections of a software system communicate amongst themselves.

Within the ClassifAI project, most of our API documentation is concerned with communication between the frontend and backend, and between the backend and the GPU server which hosts the LLM.

2.2 Assembly AI [Technology]

In the context of our project, Assembly AI was the previous machine learning model that handled text-to-speech in the previous year's iteration. Assembly AI is currently an open source machine learning project which focuses on text-to-speech, with free APIs available to anyone. It was founded in 2017 and has grown since then to become one of the development leaders in the text-to-speech space.

2.3 Backend, Frontend, GPU Server [Technology]

2.3.1 Frontend

'Backend' and 'Frontend' are two terms commonly used in the web development space. The 'Frontend' refers to any technologies, frameworks, systems, or at a lower end, code, which is sent out to end-users on a website. It is the code that makes a website look the way it does.

2.3.2 Backend

Opposite to the 'Frontend', the 'Backend' refers to any technologies, frameworks, systems, or code, which exists on a server. When an end-user requests to visit a website, that request is sent over the Internet to a server containing the matching 'Backend' code. The backend provides an intermediary for any services to be provided by a website.

2.3.3 GPU Server

For the ClassifAI project, we required a more powerful server to run our Meta Llama3 large language model. The GPU Server may be referred to in many places throughout documentation. It provides service via the backend server, but is not itself publicly accessible. Both the backend server and GPU server are located on-premises at TCU's main Fort Worth campus.

2.4 Costa's Levels of Thinking & Questioning [Education]

A framework for classifying questions, typically used in the classroom or educational setting. Questions fall into one of three categories, with 1 being more relatively surface level, and 3 being more relatively thought-provoking.

2.4.1 Costa's Level 1

Level 1 questions focus on gathering and recalling information. These questions can be answered with 'yes', 'no', or specific information found in written material, lecture material, movies, etc. A person can point to the information, read it, or physically see it.

2.4.2 Costa's Level 2

Level 2 questions focus on making sense of gathered information. These questions require students to expand what they already know by using facts, details, or clues.

2.4.3 Costa's Level 3

Level 3 questions are about applying and evaluating information. These questions require students to reflect on their thinking and be able to respond with a personal opinion that is supported by facts. The student makes a value judgment or wonders about something. There is no right or wrong answer.

2.5 Diarization [Technology]

This is the process of separating speaker from speaker in a speech-to-text transition. This is what allows the ClassifAI project to determine when the instructor is talking and when a student is talking, and also differentiates between students.

2.5.1 Pyannote's Speaker Diarization

The specific library used to implement speaker diarization in the ClassifAI project. Can be found open-source on huggingface.com, an open-source forum for machine learning related programming. It is created by Pyannote Audio.

2.6 Gemma [Technology]

Gemma is a large language model developed by Google. In regards to our project, it was the main method of question classification before upgrading to the current Meta Llama3 model.

2.7 Llama3 [Technology]

Semi-open source large language model developed by Meta. Utilized by our GPU Server to both categorize questions and summarize the transcription as a whole.

2.8 Manual Methods of Analysis [Education]

Referenced often in the project documentation as one of the main problems our project aims to solve. 'Manual methods of analysis' refers to any method in which instructors, educators, or teachers, gain insight into their own lectures. This could be via another instructor sitting in and taking notes, recording their lecture themselves and going back over it afterwards, or even collecting student feedback.

2.9 ORPO - Odds Ratio Preference Optimization [Technology]

The specific method of 'training' the language model, implemented in the ClassifAI project to train the Meta Llama3 model. The ORPO trainer is an open-source tool which facilitates training in this method. The trainer tool can be found on huggingface.com and is created by Jiwoo Hong, Noah Lee, and James Thorne.