Mamba from scratch (RUST edition)

Audric HARRIS

Chapter 2 : Creating the project

Today, we start working on the coding side of AI. The first thing to do is set up the Rust project. We will install the Rust framework and all the needed dependencies and features to allow us to make a natural language model.

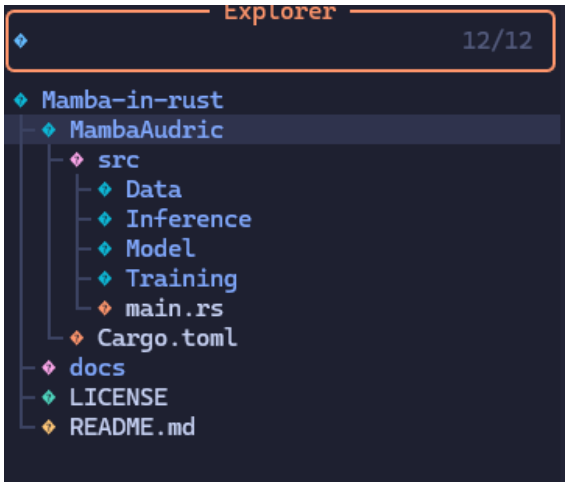
To create a project in rust we can simply use the command cargo new [name project]

In my case my project is called Mamba + my first name.

Once the project is created, we create all the folders for the project. I have separated this into 4 parts:

- **Models** (folder structure where the models are stored)
- **Data** (scripts handling the datasets we use)
- **Training** (training the AI model, separated into epoch training and reinforcement training)
- **Inference** (deploying the model and using it)

Our project is organized as follows:



We will first add the Burn crate to our project:

```
C:\Users\audri\Documents\RustTest\Mamba-in-rust\MambaAudric>cargo add burn
    Updating crates.io index
      Adding burn v0.18.0 to dependencies
             Features:
             + std
             49 deactivated features
    Updating crates.io index
     Locking 588 packages to latest Rust 1.90.0 compatible versions
      Adding matchit v0.8.4 (available: v0.8.6)
      Adding mio v1.0.4 (available: v1.1.0)
      Adding unicode-width v0.2.0 (available: v0.2.2)
```

In the screenshot, 49 features are deactivated. We will activate them throughout this project in later chapters.

The features we will be using initially are as follows:

```toml
[package]
name = "MambaAudric"
version = "0.1.0"
edition = "2024"

[dependencies]
burn = { version = "~0.18", features = ["std", "tui", "train", "vision", "wgpu", "fusion"], default-features = false }

# burn = "0.18.0"
```

Now that the project is correctly set up, we should have a solid foundation. In the next chapter, we will handle the creation of all structures that involve the models.