

Guía de Instalación para el Proyecto Final de NLP

Nombre del Autor

15 de noviembre de 2024

1. Introducción

Lamentablemente, no se ha conseguido preparar a tiempo el software necesario para los perfiles de alumno en las máquinas virtuales de Aulas Virtuales. Sin embargo, los alumnos pueden acudir físicamente al laboratorio y utilizar el usuario `usuario`, que sí tiene el software necesario. Aunque no es posible acceder a este usuario a través de Aulas Virtuales, se ha habilitado una vía alternativa para que puedan realizar la instalación del software por sí mismos. Los ordenadores han sido configurados de tal manera que los programas que instalen no se borrarán al reiniciar el equipo.

En esta guía, se incluyen las instrucciones detalladas para descargar e instalar Visual Studio Code, WSL con distribución Ubuntu y realizar una instalación de Python en Ubuntu.

2. Instalación de Visual Studio Code

2.1. Descarga

1. Visite la página oficial de Visual Studio Code: <https://code.visualstudio.com/>.
2. Haga clic en el botón *Download* y seleccione la versión para Windows.

2.2. Instalación

1. Ejecuta el archivo descargado (`VSCodeSetup.exe`).
2. Sigue las instrucciones del instalador y acepta los términos y condiciones.
3. Selecciona las opciones adicionales que queráis y completa la instalación.

Comprueba que todo funciona correctamente reiniciando el ordenador y ejecutando Visual Studio Code. En caso de que Code no aparezca, avisad a STIC en el CAU o a Jaime Pizarroso.

2.3. Extensiones

Ya debería haber ciertas extensiones instaladas. En caso contrario, las extensiones que necesitáis son:

1. Python
2. Pylance
3. Python Debugger
4. La suite de Remote (incluido WSL)
5. Jupyter
6. Jupyter Cell Tags
7. Jupyter Notebook Renderers
8. Jupyter Slide Show

3. Instalación de WSL con Ubuntu

3.1. Activación de WSL

1. Actualiza el kernel de wsl visitando <https://aka.ms/wsl2kernel> o descargando el paquete de actualización directamente de aquí.
2. Ejecuta el archivo `wsl_update_x64.msi` e instala la actualización.
3. Define wsl 2 como versión predeterminada con el comando:
`wsl --set-default-version 2`
4. Descarga el archivo de instalación de Ubuntu para WSL de este link: <https://aka.ms/wslubuntu2204>. Tarda unos cinco minutos en descargarse.
 - No descarguéis Ubuntu desde Microsoft Store, ya que STIC lo tiene restringido y no os dejará descargárselo.
5. Se habrá descargado un archivo llamado “Ubuntu2204-221101.AppxBundle”. Renombrar este archivo como “Ubuntu2204-221101.zip” y extraerlo en una carpeta aparte.
6. Ir a la carpeta en la que hayáis extraído los archivos. En dicha carpeta deberían aparecer los ficheros como en la figura 1.
7. Abre un ventana de Powershell y cambia el directorio a la carpeta con los ficheros de la figura 1. Ejecuta el siguiente comando para instalar la distribución de Ubuntu:
`Add-AppxPackage .\Ubuntu_2204.1.7.0_x64.appx`

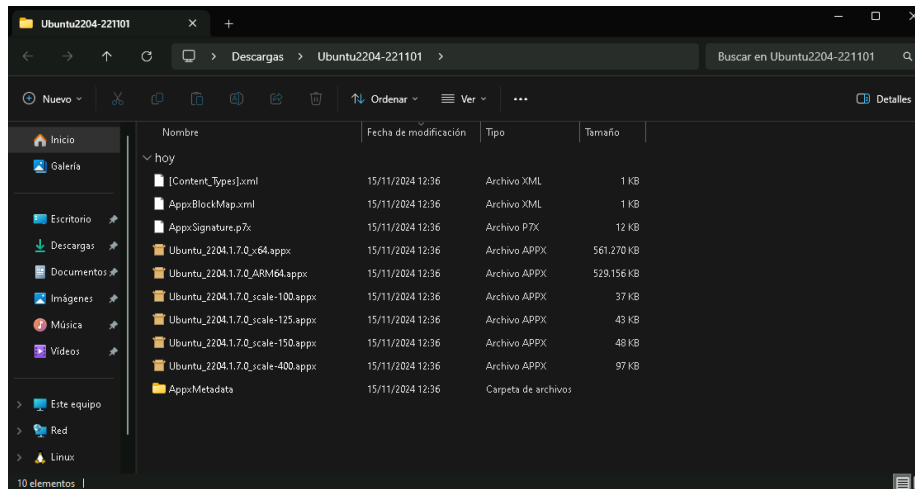


Figura 1: Ficheros extraídos del archivo “Ubuntu2204-221101.AppxBundle”

8. Ahora hay que añadir la nueva distribución al PATH de windows con el siguiente comando:

```
$userenv = [System.Environment]::GetEnvironmentVariable("Path","User")

[System.Environment]::SetEnvironmentVariable("PATH", $userenv
+ ";C:\Users\<user>\Ubuntu","User")
```

9. Estable la distribución de Ubuntu como la de por defecto de WSL:

```
wsl --set-default Ubuntu
```

Comprueba que Ubuntu se ha instalado correctamente abriendo el menú de Inicio y ejecutando Ubuntu.

3.2. Configuración Inicial

1. Ejecuta *Ubuntu* desde el menú de inicio.
2. Espera a que se complete la configuración inicial.
3. Crea un nombre de usuario y contraseña cuando se le solicite. Recuerda esta contraseña porque hará falta para ejecutar comandos como administrador.

4. Instalación de Python 3.10 en Ubuntu

1. Abra el terminal de Ubuntu.
2. Actualice la lista de paquetes:

```
sudo apt update
```

3. Actualice las versiones de los paquetes:

```
sudo apt upgrade
```

4. Actualice las librerías necesarias para instalar python:

```
sudo apt install build-essential zlib1g-dev libncurses5-dev  
libgdbm-dev libnss3-dev libssl-dev libsqlite3-dev libreadline-dev  
libffi-dev curl libbz2-dev lzma liblzma-dev libbz2-dev
```

5. Descargue el fichero Python 3.10.8:

```
wget https://www.python.org/ftp/python/3.10.8/Python-3.10.8.tgz  
tar -xzf Python-3.10.8.tgz
```

6. Ejecuta el fichero de configuración:

```
cd Python-3.10.8  
./configure --enable-optimizations
```

7. Construye el fichero de instalación e instálalo:

```
make -j 2  
sudo make install
```

8. Cambia el alias de python para facilitar su uso:

```
alias python='/usr/local/bin/python3.10'  
source ~/.bashrc
```

9. Verifica la instalación de python ejecutando:

```
python --version
```

10. Instala la herramienta pip:

```
sudo apt install python3-pip
```

11. Finalmente, tus profesores han dejado un fichero *requirements.txt* en Moodle. Instálalo usando:

```
pip install -r requirements.txt
```

5. Creación de un token de huggingface

Para descargar algunos modelos y datasets de huggingface, vas a necesitar estar autenticado en el ordenador. Para ello:

1. Ve a <https://huggingface.co/> y loggeate en tu cuenta. Si no tienes cuenta, crea una nueva.
2. En el icono de tu cuenta (esquina superior derecha), accede al menú de *Access tokens* (ver figura 2).

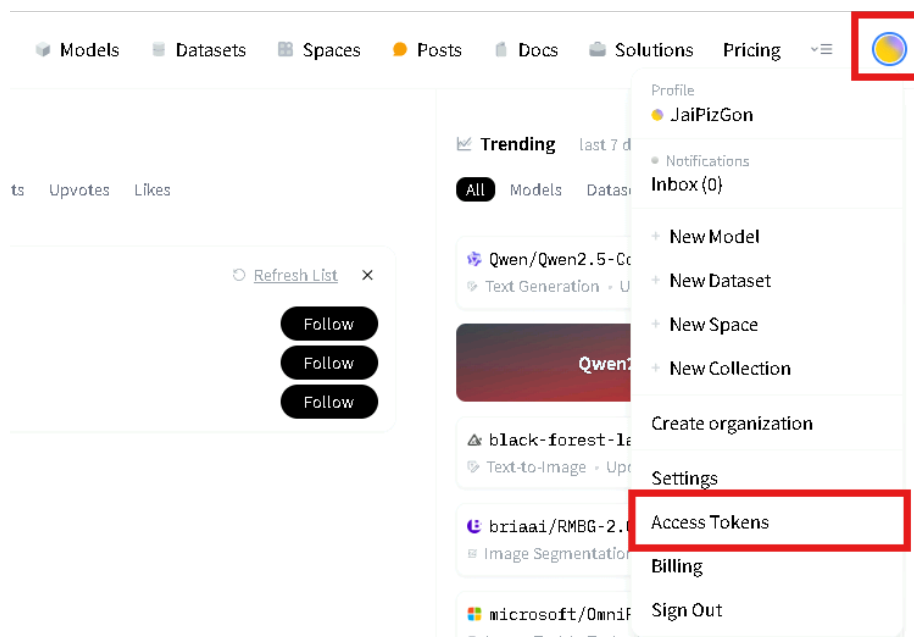


Figura 2: Menú de tokens de acceso de huggingface

3. Crea un nuevo token haciendo click en el botón *Create a new token* (ver figura 3).
4. Crea un nombre para tu nuevo token de acceso (por ejemplo, “nlp-final-project”) y dale permisos para leer y escribir en repositorios de huggingface (ver figura 4). Baja hasta el final de la página web y haz click en el botón “Create token”.
5. Aparecerá un ventana modal con la clave alfanumérica del token que tendrás que usar a posteriori para autenticarte en Python. Copia esta clave en un fichero de texto o en algún sitio donde no la pierdas.
6. Abre una terminal de comandos de WSL en la distribución de Ubuntu que hemos instalado antes.

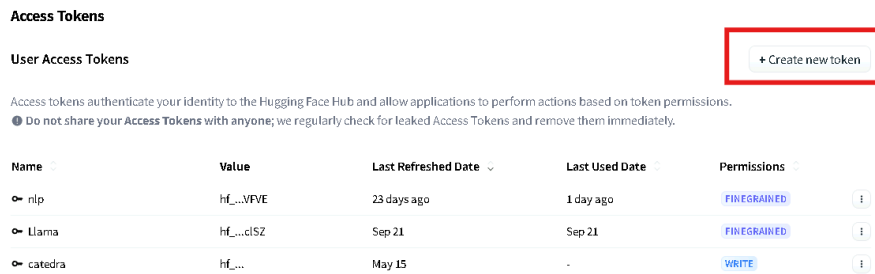


Figura 3: Botón de *Create new token*

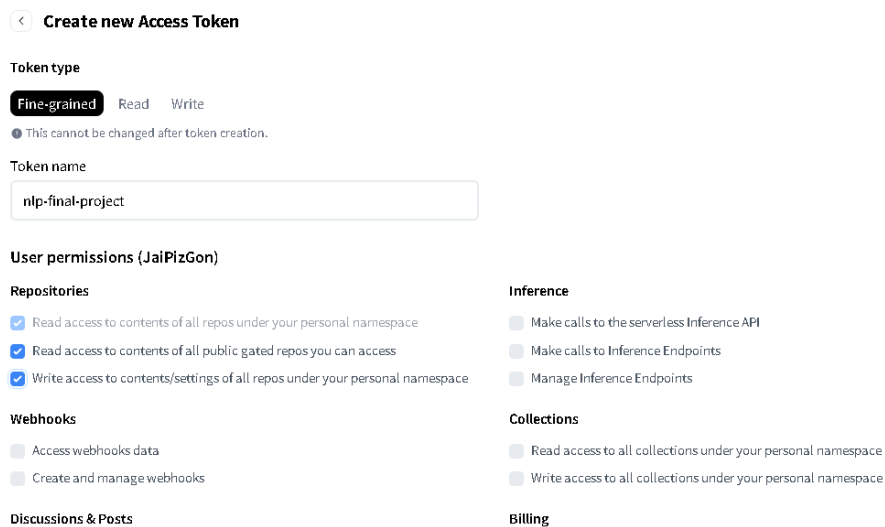


Figura 4: Nombrar y dar permisos al token.

7. Ejecuta el siguiente comando para comenzar el proceso de autenticación (ver figura 5):
- ```
huggingface-cli login
```

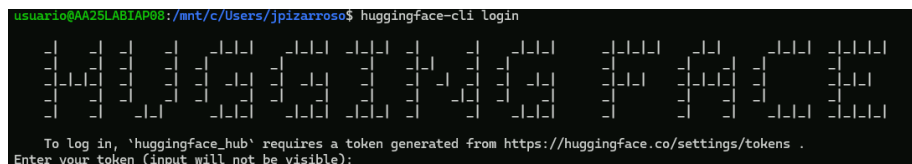


Figura 5: Proceso de autenticación de huggingface

- Después de copiar y pegar el token que has creado anteriormente, ya podrás descargarte datasets y modelos que necesiten identificación.

## 6. Comprueba que todo funciona

Descárgate el starter kit de la práctica final e intenta ejecutar la primera celda del notebook “model\_finetune\_and\_compression.ipynb” estando conectado al WSL de Ubuntu (ver figura 6). Te recuerdo que para ir a la carpeta Downloads estando conectado a WSL, la ruta del archivo será algo parecido a “/mnt/c/Users/jpizarroso/Downloads/starter-kit/”. The first time you do this, VS Code might require you to install the Python extension and the jupyterhub extensions again, and the ipykernel package.



Figura 6: Comprobación de conexión de VS Code a WSL.

## 7. A word of advice

Tened cuidado con la memoria que ocupáis con los modelos y los datasets, ya que cada modelo ocupa del orden de unos 15GB aproximadamente en disco. Si vais a probar varios modelos, acordaros de borrar los modelos y datasets que no utilizéis de la caché de huggingface utilizando el comando `huggingface-cli delete-cache` en una terminal de wsl y siguiendo las instrucciones.