Python DSI_BERT Tutorial

This tutorial covers how to set up and use a local Python version of the DSI algorithm (using BERT as a back end). Disclaimer: DSI_BERT is fairly resource-hungry; it is recommended to run this on a system with at least 4 logical cores and 8GB of RAM. A troubleshooting section is included at the bottom of the tutorial.

PART 1: DOWNLOAD FILES AND CREATE THE DSI DIRECTORY

- Download 'DSI.py,' 'environment.yml,' and the tutorial sample data, 'Study1_OSF Final.csv.'

- Create a new folder/directory on your machine somewhere at the user level named 'DSI'. Put all three files into the DSI folder.

PART 2: INSTALL MINICONDA AND CLONE THE DSI VIRTUAL ENVIRONMENT

- Install Miniconda for your appropriate system using this link. Be sure to install it at the *user* and not system level on your machine if using a university computer. Miniconda is a Python virtual environment and package manager.

- Once downloaded, open Terminal (Mac/Unix) or Command Prompt (Windows). Navigate to the DSI folder in your terminal by typing 'cd' followed by a single space and then dragging and dropping the 'DSI' folder from your file explorer into the command line. A file path should now be visible in the terminal. Then hit Enter and you will be in the 'DSI' directory. If this does not work, see the troubleshooting section.

- Once in the DSI directory, type `conda env create -f environment.yml` in your terminal and hit Enter to clone the DSI virtual environment. This will create a virtual environment called 'dsi' and will install the appropriate versions of python and all required packages. Leave this terminal open (in the DSI directory), as we'll use it in the next part.

PART 3: RUN DSI

- If you wish to run the DSI algorithm on your own data, add your story-containing .csv file to the DSI directory. The file should be organized with one row per participant. The file can contain additional columns but *must* contain a column named **ID** (the participant identification number) and another called **story** (which contains the participant's entire passage). The ID/story column names are case sensitive and cannot have leading or trailing spaces. Ensure the data in your story columns matches the format seen in the included sample dataset ('Study1_OSF Final.csv'). If *not* using pyDSI on your own data, you can proceed through the tutorial using the included 'Study1_OSF Final.csv' dataset.

- The 'DSI.py' script is set up by default to run DSI on the included sample data—no editing necessary. If you wish to run the script on your own data, open 'DSI.py' with a text editor of your choice. Navigate to the USER EDIT section, change the value of the 'filename' variable to the name of your .csv, and save.
  - The edited line should look like this: `filename = 'your_input_data.csv'`
  - Note: the input file must be a .csv (not an .xlsx or .xls file), so resave your data as a .csv beforehand if need be.
  - If you would like to do a small scale test that performs DSI only for the first participant/story, un-comment the second line in the LOAD DATA section of DSI.py (i.e., delete the # and any leading spaces).

• With your command line interface, activate the 'dsi' virtual environment by typing `conda activate dsi` and hit Enter. You should now see '(dsi)' prepended to the command line.

• Finally, run the DSI Python script by typing `python DSI.py` into your terminal and hitting Enter. This may take a while, depending on your system specifications. On a system with 24 logical cores and 32GB of RAM, the script will compute DSI values for all 180 participants in 'Study1_OSF Final.csv' in approximately 6 minutes. When the script finishes, it will save a file in the DSI directory called 'DSI-output.csv'. The file will contain all of the data contained in the input .csv plus a new column **DSI** that contains DSI values.

TROUBLESHOOTING

- ***Navigation to the DSI folder via the command line (PART 2, 2nd bullet)***
  - If your machine does not let you change to the DSI directory in the terminal, or if the terminal prints 'access denied', security/permissions are likely the problem.

    - If on a personal computer:
      - Mac/Unix: Permissions issues are unlikely to affect this part for these users.
      - Windows: Try opening Command Prompt as an administrator by right clicking the Command Prompt app in the search menu and selecting 'Run as Administrator'. Then, try changing directories again. If this does not work, try the instructions below for university computers.

    - If on a university computer:
      - Mac/Unix: Ensure the DSI folder is saved at the user and not system level (e.g., Documents folder). If this does not work, you will need to request administrator rights to your computer from your university's IT services.
      - Windows: First, ensure the DSI folder is saved in a place you have permissions to at the user (not system) level and in a drive you have access to (likely the C: drive). From the Windows start menu, scroll down until you find the Anaconda folder. Click the dropdown arrow and open 'Anaconda Powershell Prompt (miniconda3)'. Try using this terminal to complete PART 2 (2nd bullet). If this does not work, you will need to request enhanced permissions or administrator rights to your machine from your university's IT services.

- ***Cloning or activating the dsi virtual environment (PART 2.3 or PART 3.3)***
  - If on a personal computer:
    - Mac/Unix: If an error occurs, try running the conda command as root—i.e., 'sudo conda env create -f environment.yml'. It will ask you for your computer's password.
    - Windows: If an error occurs, try opening Command Prompt as an administrator (right click the Command Prompt app in the Windows start menu and select 'Run as Administrator') and running the conda command again.

  - If on a university computer:
    - Unix & Windows: Ensure you downloaded miniconda at the user, rather than system, level. If not, uninstall and reinstall at the user level and in a location/drive your user account has access to. If this does not work, you will need to request enhanced permissions or administrative rights for your computer from the university's IT services.

- ***Running DSI.py (PART 3)***

  - If running 'python DSI.py' at the command line results in a Permission denied error:
    - Personal computer: Run the script with enhanced permissions. On Mac/Unix, enter the command 'sudo python DSI.py' and enter your computer's password. On Windows, open a Command Prompt as an administrator by right clicking the Command Prompt app in the Windows start menu and clicking 'Run as administrator'.
    - University computer: Enhanced permissions and/or administrative rights to your machine are needed; reach out to your university's IT services to request this.

  - Non-permission-based errors, after running 'DSI.py':
    - If the terminal returns an error saying it cannot find your input .csv file, make sure: (1) the intended input .csv is in the 'DSI' folder, (2) the filename variable in 'DSI.py' exactly matches the intended input .csv's name (in a case-sensitive way), and (3) that the input file is in .csv format. Additionally, check that the edited line—i.e., filename = 'your_input_data.csv'—is left aligned with no preceding spaces.
    - If all of the above is in order, there may issues reading in some of your .csv's data. Try removing all columns besides the ID and story columns, saving, and running 'python DSI.py' again.
    - If an error indicates python cannot find the ID and/or story columns, check that those columns are included in your input .csv and that the case of those column names matches that shown in the sample dataset. Additionally, be sure there are no leading or trailing spaces surrounding the ID/story column names. Resave your .csv with these edits if needed and try running 'python DSI.py' in the terminal again.