

Manual TNT Matrix Builder 1.0

Desarrollado por: Jonathan Liria

TNT Matrix Builder es una aplicación gráfica desarrollada en Python diseñada específicamente para facilitar la enseñanza y el aprendizaje del software TNT en cursos de Sistemática Filogenética y Cladística. El programa aborda uno de los mayores desafíos para los estudiantes e investigadores: la integración de "evidencia total" (datos moleculares, morfológicos discretos y caracteres continuos) en un único archivo de análisis. Si bien paquetes como Mesquite (Maddison & Maddison, 2025) son excelentes para la edición visual de múltiples matrices, carecen de una función de exportación combinada que automatice la sintaxis requerida por TNT. Esto obliga al usuario a recurrir a una edición manual de texto, un proceso que involucra la configuración precisa de comandos como `nstates`, `xread` y etiquetas de bloque (`&[dna]`, `&[num]`, `&[cont]`), los cuales suelen ser fuente de errores frecuentes durante las prácticas.

Esta herramienta complementa y expande el flujo de trabajo de aplicaciones previas como `py_tm2tnt` y `py_tps2tnt` (Liria & Soto-Vivas, 2025), permitiendo que estudiantes o investigadores que se inician en el uso de TNT puedan realizar análisis filogenéticos que integren un enfoque integrativo. Al automatizar la concatenación de múltiples bloques de datos, TNT Matrix Builder elimina posibles errores en la manipulación del archivo `.tnt`, permitiendo que el usuario se concentre en el análisis filogenético como tal. El programa no solo sincroniza automáticamente los nombres de los taxones mediante protocolos de unión e intersección, sino que también gestiona la complejidad de los datos de morfometría geométrica 2D (archivos TPS), promediando especímenes por taxón e integrando escalas de medición según las directrices de Catalano & Goloboff (2018). Esta herramienta se presenta, por tanto, como un puente tecnológico esencial que simplifica la transición desde la recolección de datos heterogéneos hasta su procesamiento en el motor de búsqueda filogenética de TNT (Goloboff & Morales, 2023).

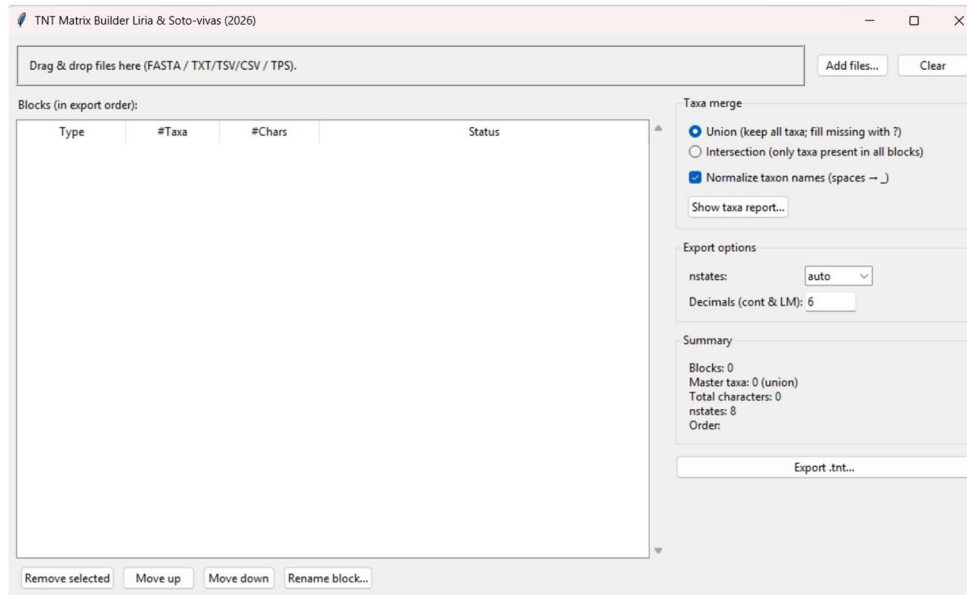


Figura 1. Vista general de la interfaz de TNT Matrix Builder mostrando el área de "Blocks" y el panel de configuración lateral.

1. Requisitos

Antes de utilizar **TNT Matrix Builder**, asegúrese de tener instalado:

- **Python 3.x**
- **Librerías estándar:** `tkinter`, `re`, `math`, `os`, `dataclasses`.
- **Librería opcional:** `tkinterdnd2` (para habilitar la función de arrastrar y soltar archivos). Puede instalarla con el comando: `pip install tkinterdnd2`.

2. Formatos soportados

El programa detecta automáticamente la naturaleza de los datos basándose en su contenido:

- **Molecular (DNA):** Archivos FASTA (`.fasta`, `.fa`, `.fas`).
- **Tables (`.txt`, `.tsv`, `.csv`):**
 - * **Discrete/Numeric:** Si el archivo contiene enteros o polimorfismos.
 - **Continuous:** Detectado automáticamente por la presencia de valores decimales.
- **Morphometrics (TPS):** Archivos `.tps` con landmarks 2D. El programa promedia automáticamente los especímenes por especie.

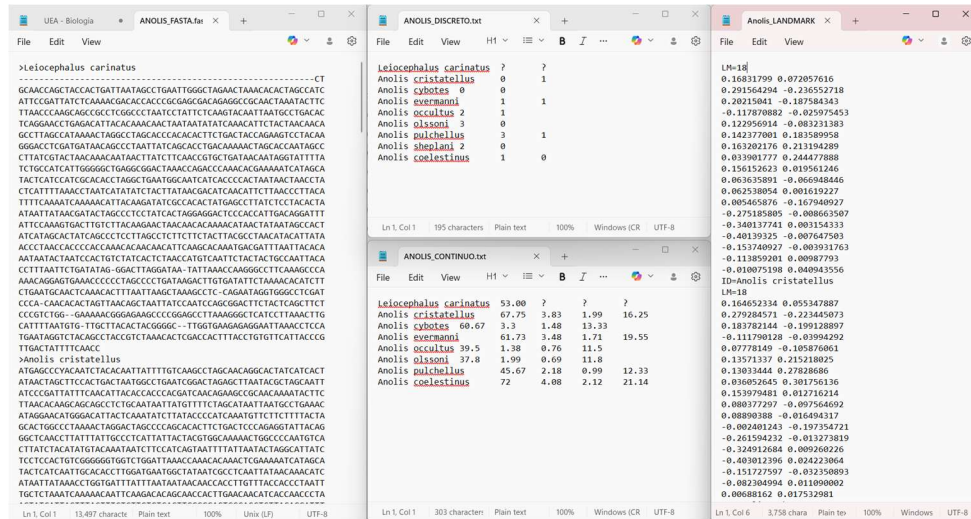


Figura 2. Ejemplo de los varios tipos de datos soportados: secuencias FASTA, tablas de datos continuos/discretos y archivos TPS de morfometría.

3. Guía paso a paso

Paso 1: Carga de archivos. Añada sus matrices mediante el botón **"Add files..."** o arrastrando los archivos directamente al panel superior de la interfaz. Los bloques aparecerán listados con su tipo de dato detectado automáticamente.

Paso 2: Organización y regla de prioridad. Esta opción permite organizar el orden de los bloques en TNT; por ejemplo, se desea que los caracteres moleculares (COI, 16s, etc.) estén al final, y los otros (continuos, discretos) al inicio. Si el proyecto incluye caracteres **continuos** o **Landmarks**, estos deben colocarse siempre en la primera posición de la lista (Bloque 1). Utilice los botones **"Move up"** y **"Move down"** para ajustar el orden. Puede usar **"Rename block..."** para asignar nombres descriptivos para cada bloque que aparecerán en la sección `cname`.

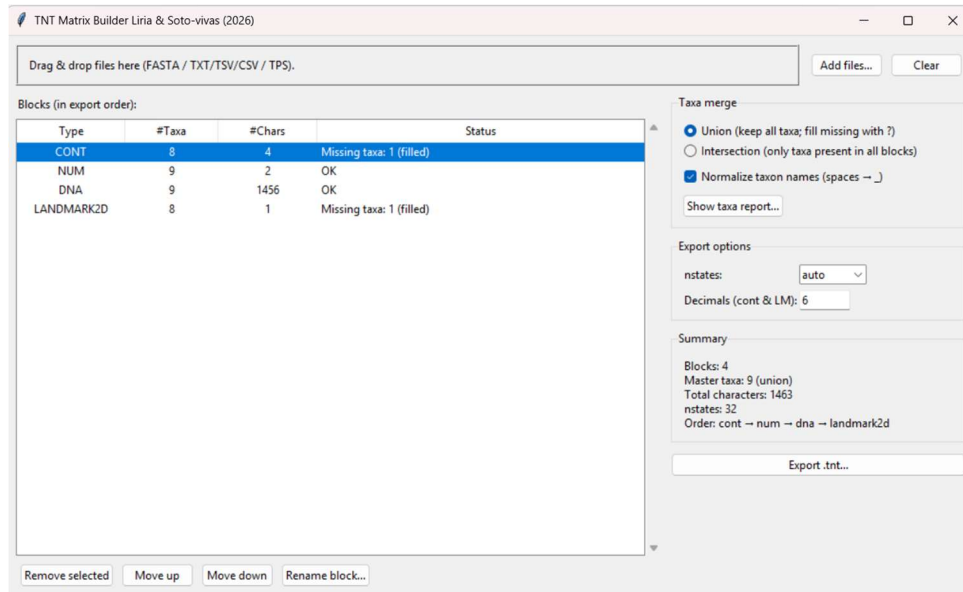


Figura 3. Detalle de la lista de bloques indicando el uso de los botones de movimiento para posicionar los datos continuos al inicio.

Paso 3: Concatenación de taxones (Taxa merge).

- **Union:** Mantiene todos los nombres de taxones y rellena los bloques faltantes con datos vacíos (?).
- **Intersection:** Solo mantiene los taxones que están presentes en todos los archivos cargados.
- **Normalize names:** Reemplaza espacios por guiones bajos (_) para terminales en donde se asigna Género y especie, requisito en TNT y fuente común de errores.

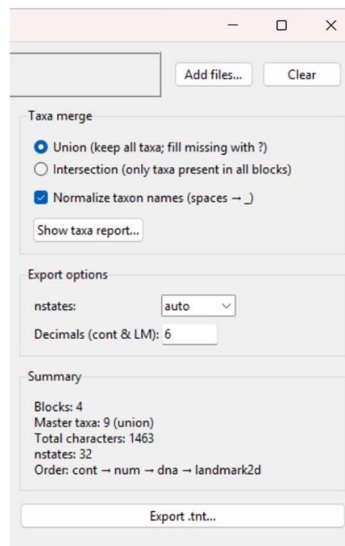


Figura 4. Panel de opciones de "Taxa merge" con las funciones de Union/Intersection y normalización de nombres de especies.

Paso 4: Exportación de la matriz a TNT. Seleccione el rango de estados en "**nstates**" (el programa elegirá 32 automáticamente si detecta datos continuos). Si las matrices concatenadas solo poseen datos moleculares, se recomienda elegir **nstates 8**, para optimizar el uso de memoria. Finalmente, haga clic en "**Export .tnt...**" para obtener su archivo concatenado.

4. Reporte de concatenación

Antes de exportar, se recomienda consultar el botón "**Show taxa report...**". Esta función permite validar la integridad de la matriz, mostrando qué especies están ausentes en cada bloque y confirmando el número total de caracteres que tendrá el análisis final.

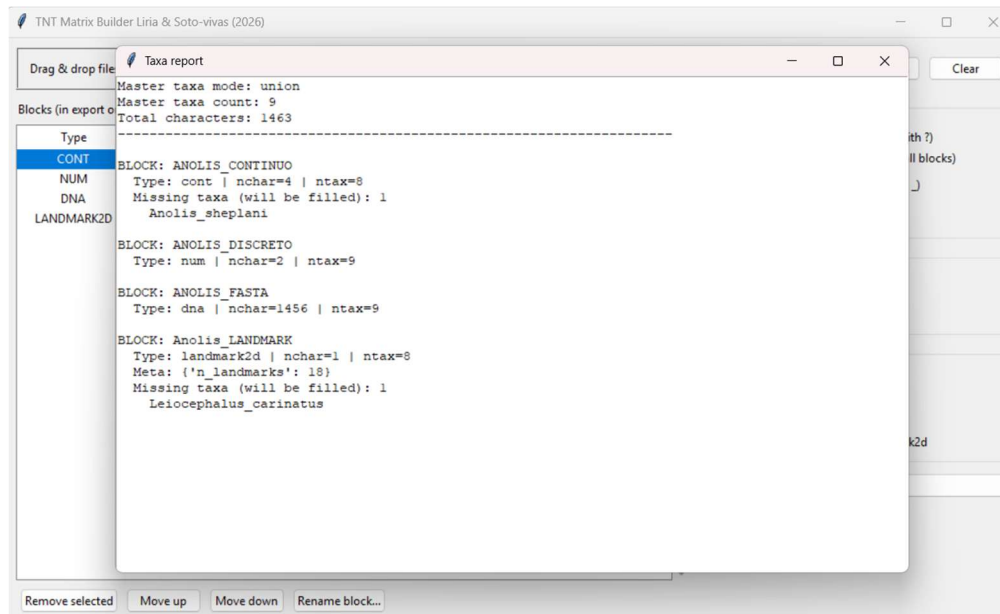


Figura 5. Ventana de reporte de taxones ("Taxa report") mostrando el listado de especies ausentes por cada archivo fuente cargado.

5. Referencias

- Catalano, S., Goloboff, P.A. (2018). *A guide for the analysis of continuous and landmark characters in TNT*.
- Goloboff, P.A., Morales, M.E. (2023). *TNT version 1.6, with a graphical interface for MacOS and Linux, including new routines in parallel*. Cladistics, 39: 144-153.
- Liria, J., Soto-Vivas, A. (2025). *py_tps2tnt y py_tm2tnt: dos programas en Python para procesamiento de datos morfométricos en análisis cladísticos con TNT*. Revista Peruana de Biología, 32(2), e30018.
- Maddison, W. P., D.R. Maddison. (2025). *Mesquite: a modular system for evolutionary analysis. Version 4.02*. <https://www.mesquiteproject.org>

