Procesamiento Masivo de Datos

Alex Di Genova

August 18, 2025

Universidad de O'higgins

Outline

Bienvenida curso de PMD

Presentación

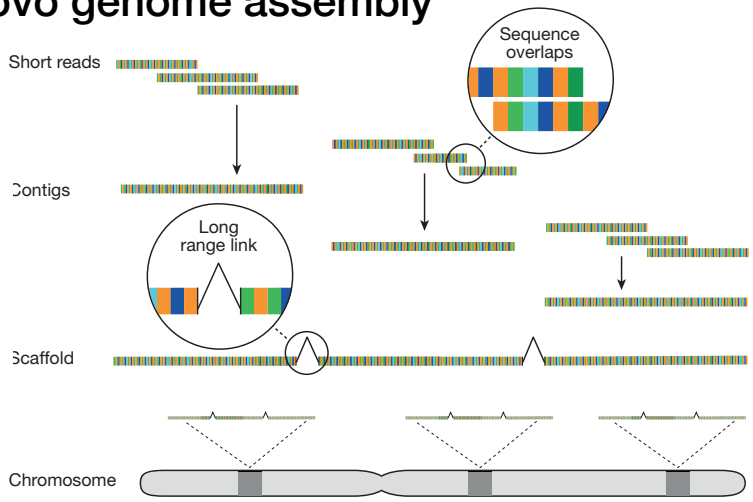
Alex Di Genova

- 2003–2008 Ingeniero en Bioinformática.
- 2013-2017 Doctor en Sistemas Complejos.
- 2017-2021 Postdoctorado en algoritmos y cáncer (Francia).
- 2022-2023 Profesor Asistente UOH.
- 2023-Presente Profesor Asociado UOH.
    - Di Genoma Lab
        - Combinamos el desarrollo de nuevos algoritmos, análisis de genomas y tecnologías ómicas de última generación para estudiar sistemas biológicos complejos.

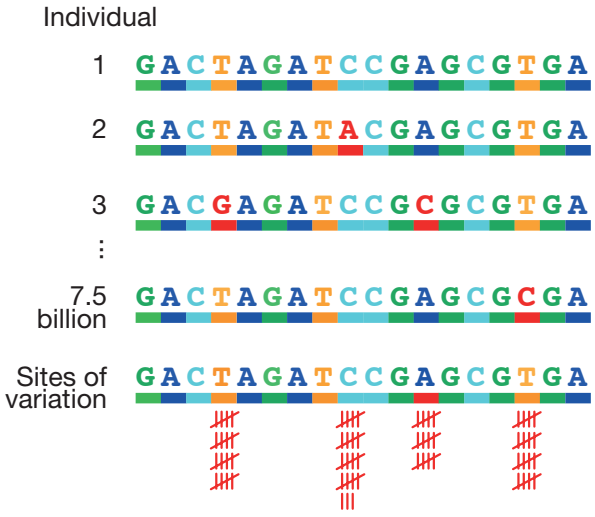
Ejemplos donde se usa PMD

Sequencing technologies

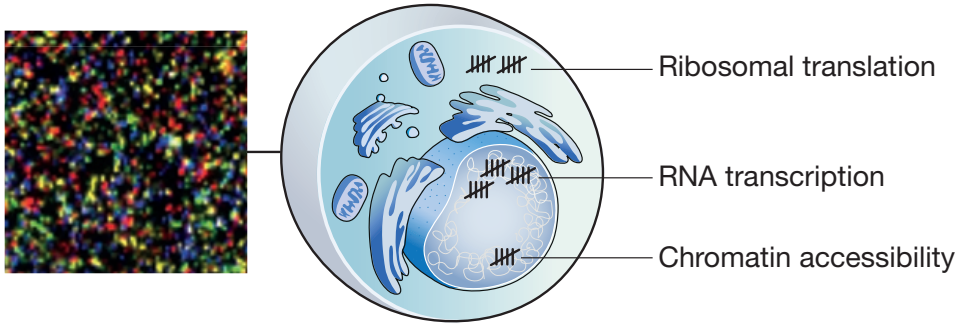
De novo genome assembly



Genome resequencing

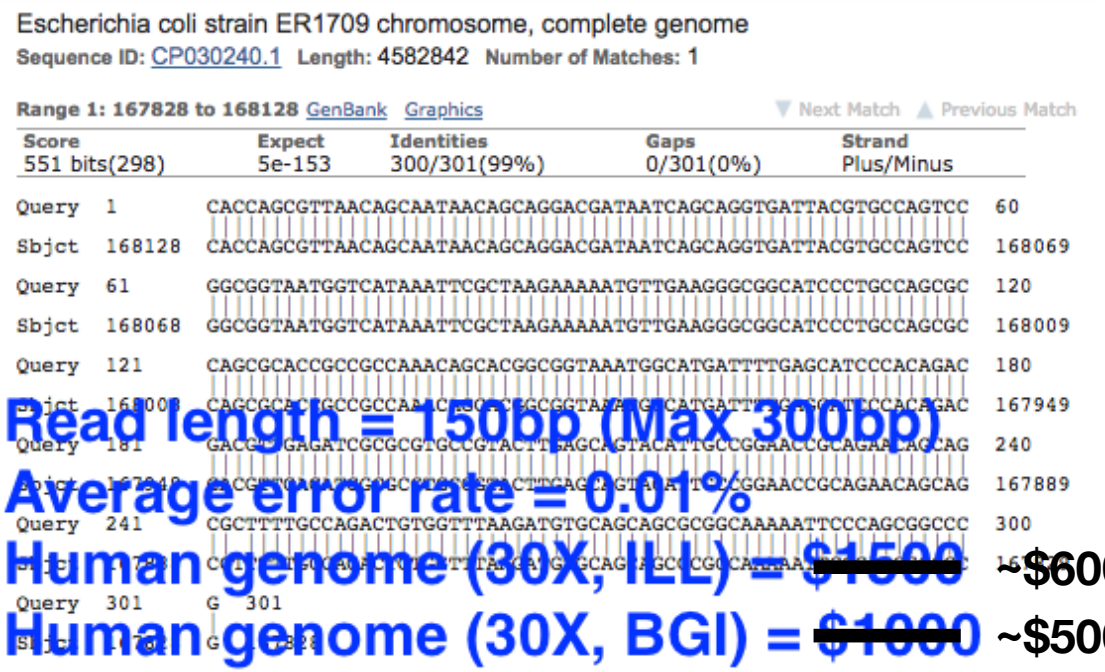


Sequencer as counting devices



Shendure, Jay, et al. DNA sequencing at 40: past, present and future. *Nature* 550.7676 (2017):

Sequencing technologies



Read length = 150bp (Max 300bp)

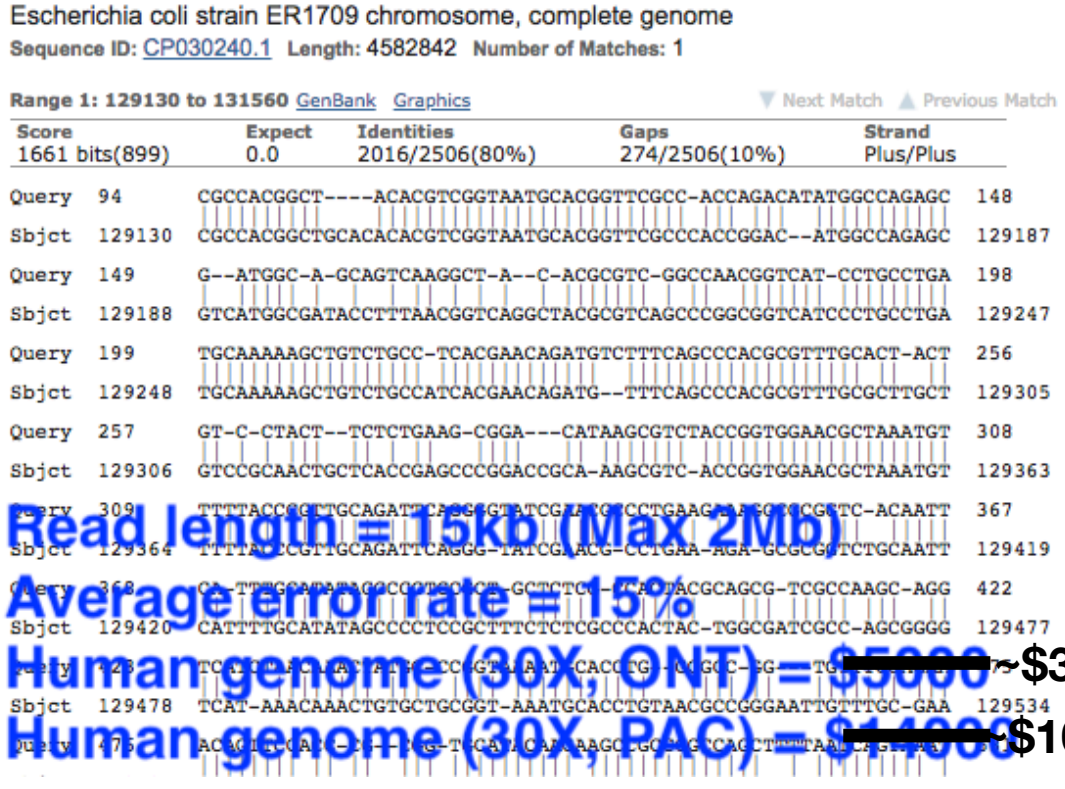
Average error rate = 0.01%

Human genome (30X, ILL) = ~~$1500~~ ~$600

Human genome (30X, BGI) = ~~$1000~~ ~$500

Short

Long



Read length = 15kb (Max 2Mb)

Average error rate = 15%

Human genome (30X, ONT) = ~~$5000~~ ~$3000

Human genome (30X, PAC) = ~~$14000~~ ~$10000

Genome assembly

Genome

Sequencing technology

Reads

Assembler

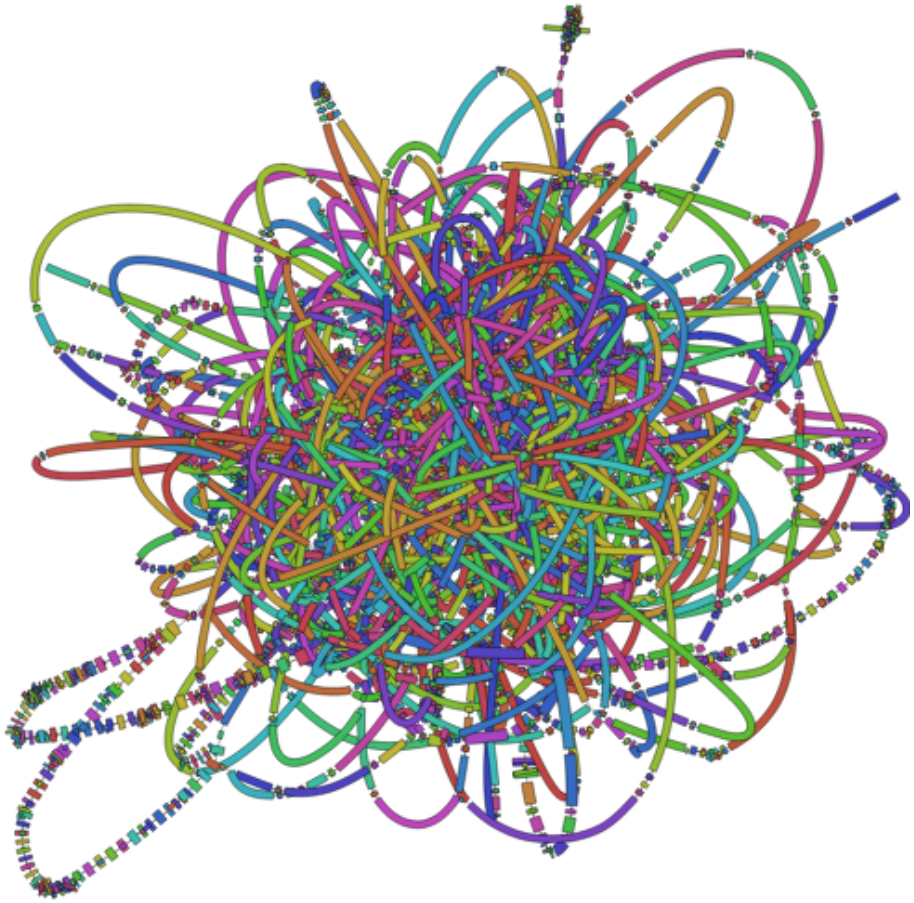
Overlaps

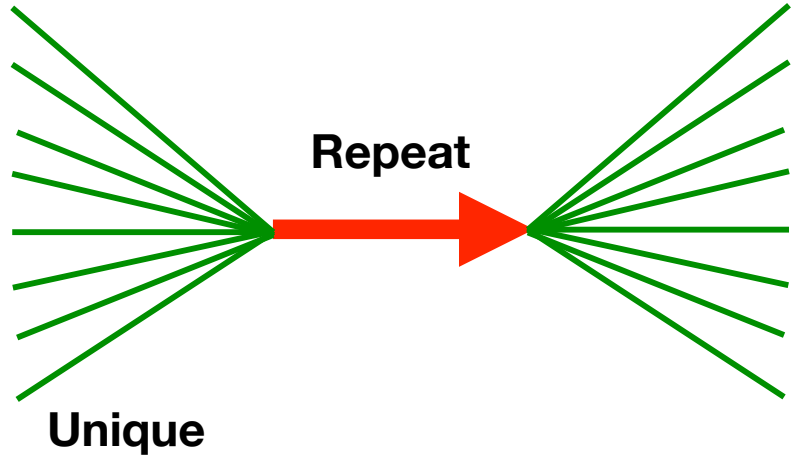
Assembly graph

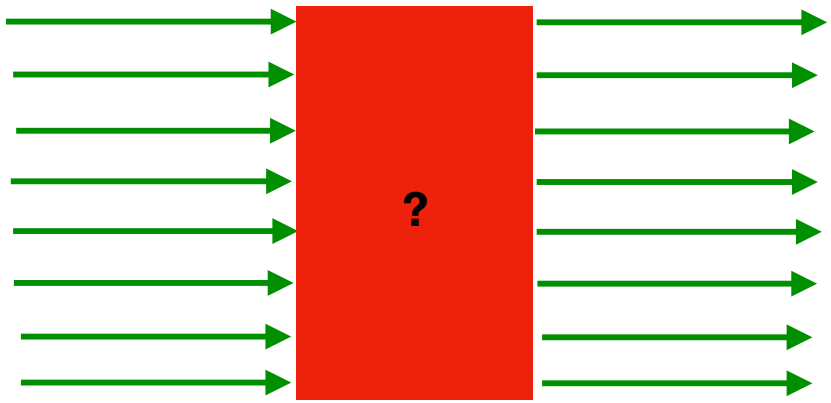
The genome path (sequence)

40 years of genome assembly

Genome assembly is complex





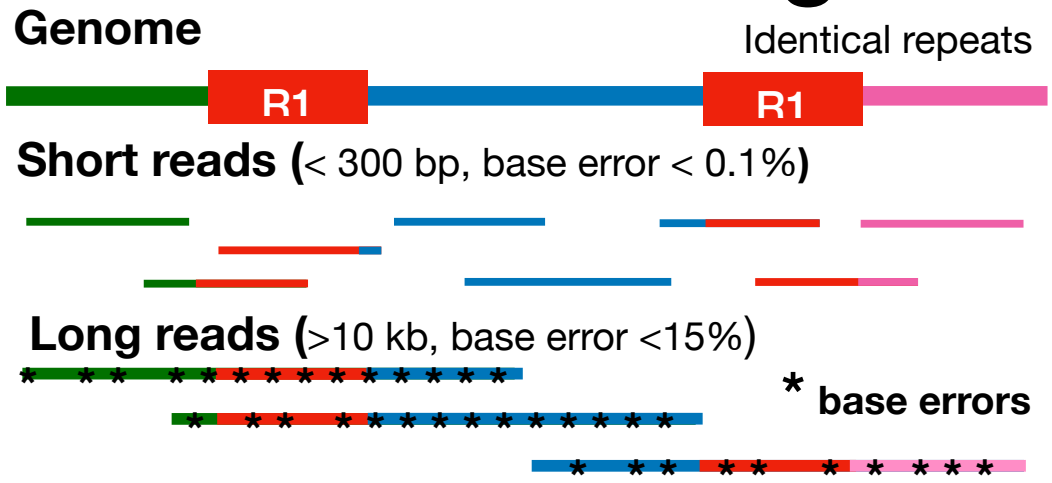


How do you get through?

How do we make “the” genome path?

Hybrid assembly: How can we combine short and long reads?

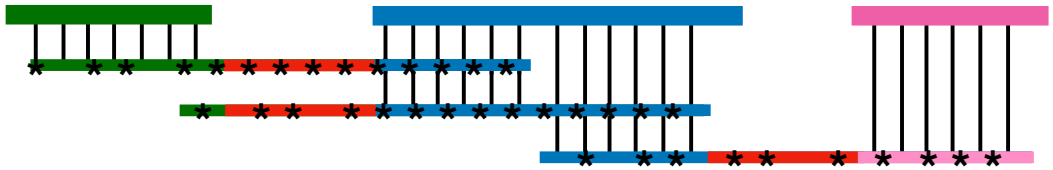
Sequencing technology



Wengan Assembler

(1) Assemble short-reads

(2) Scaffold using long reads

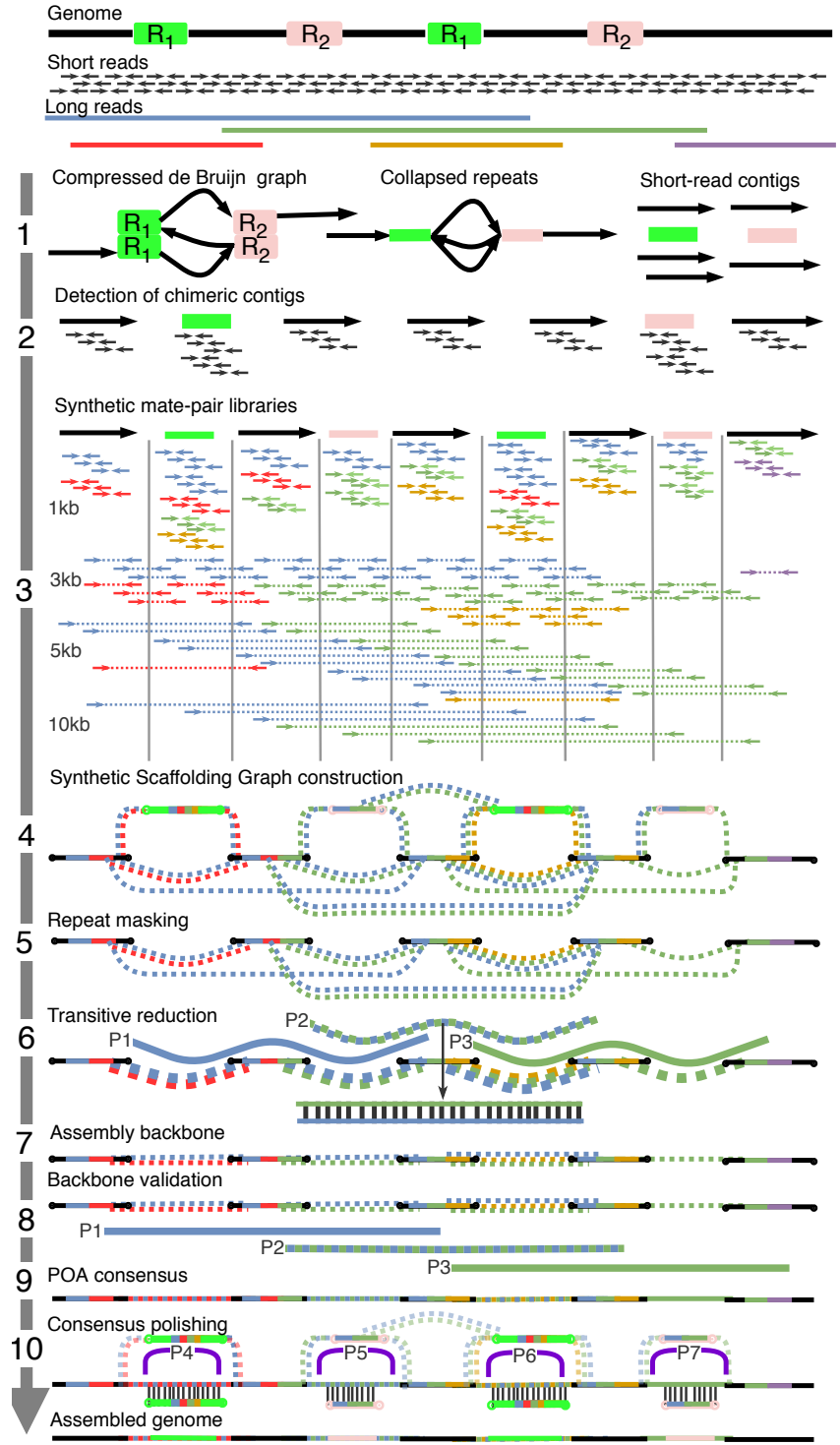


(3) Refine repetitive regions (polishing)

The resulting assembly is both *contiguous and accurate*

Wengan: a new assembly paradigm

- **Full** hybrid assembler.
- Avoids entirely all-vs-all read comparisons (**fast**).
- A new assembly graph (*GoogleMaps*).
- 1.5 years of development.
  - ~20k lines of code (C++, PERL)
- https://github.com/adigenova/wengan



- **Di Genova, A**. (2018). Fast-SG: an alignment-free algorithm for hybrid assembly. *GigaScience*, *7*(5).
- **Di Genova, A**. (2021). Wengan: Efficient and high-quality hybrid de novo assembly of human genomes. *Nature Biotechnology*.

Wengan

ARTICLES
https://doi.org/10.1038/s41587-020-00747-w

nature biotechnology

Check for updates

OPEN

Efficient hybrid de novo assembly of human genomes with WENGAN

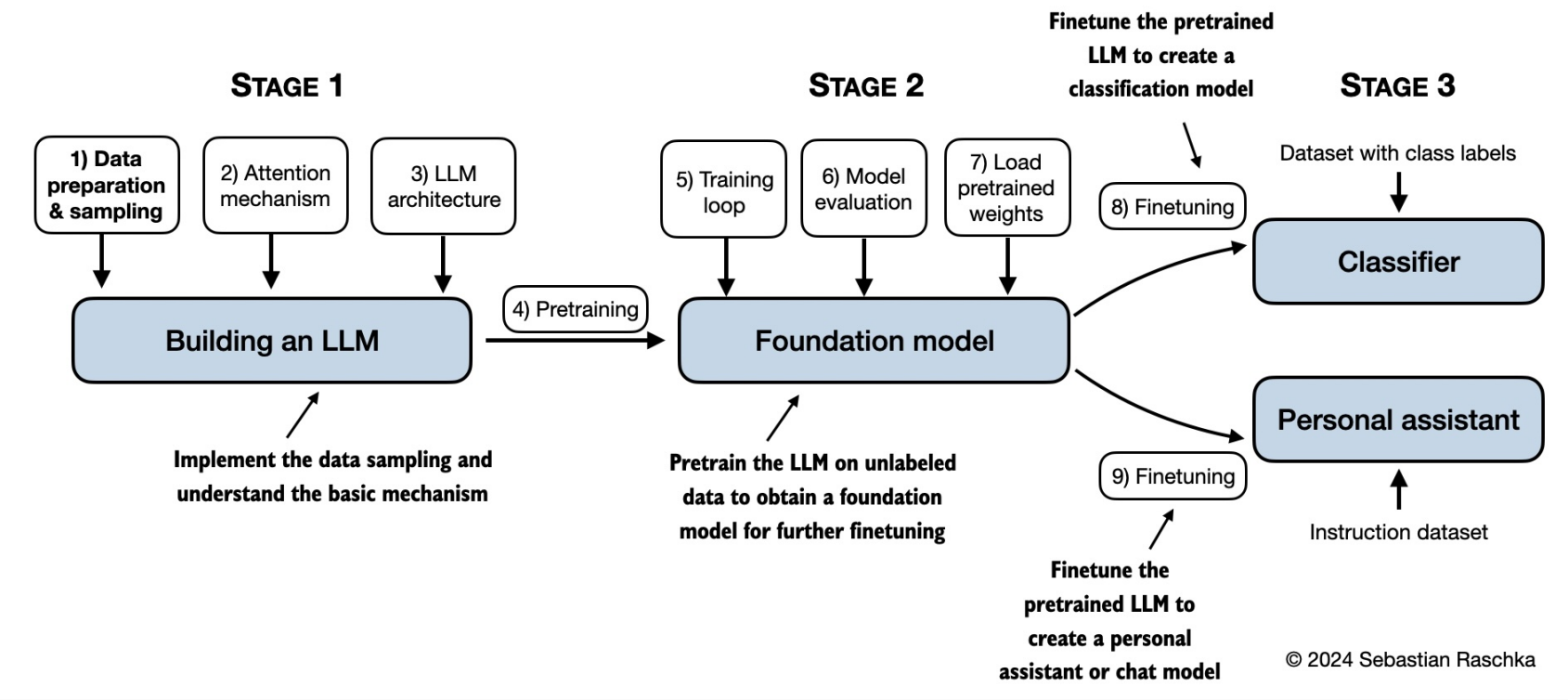
Alex Di Genova[1,2] ✉, Elena Buena-Atienza[3,4], Stephan Ossowski[3,4] and Marie-France Sagot[1,2] ✉

Generating accurate genome assemblies of large, repeat-rich human genomes has proved difficult using only long, error-prone reads, and most human genomes assembled from long reads add accurate short reads to polish the consensus sequence. Here we report an algorithm for hybrid assembly, WENGAN, that provides very high quality at low computational cost. We demonstrate de novo assembly of four human genomes using a combination of sequencing data generated on ONT PromethION, PacBio Sequel, Illumina and MGI technology. WENGAN implements efficient algorithms to improve assembly contiguity as well as consensus quality. The resulting genome assemblies have high contiguity (contig NG50: 17.24–80.64 Mb), few assembly errors (contig NGA50: 11.8–59.59 Mb), good consensus quality (QV: 27.84–42.88) and high gene completeness (BUSCO complete: 94.6–95.2%), while consuming low computational resources (CPU hours: 187–1,200). In particular, the WENGAN assembly of the haploid CHM13 sample achieved a contig NG50 of 80.64 Mb (NGA50: 59.59 Mb), which surpasses the contiguity of the current human reference genome (GRCh38 contig NG50: 57.88 Mb).

WENGAN is a Mapudungun word.

WENGAN means "Making the path".

Entrenando un LLM



Planificación curso PMD

Unidades

- 4 Unidades (15 semanas)
  - Distribución y Paralelismo (4 semanas)
  - Modelamiento de Procesamiento Distribuido (5 semanas)
  - Modelos de Almacenamiento Escalable (3 semanas)
  - Bases de datos Distribuidas (3 semanas)

Evaluaciones

- Controles (75%, potenciales fechas de controles):
  - Control 1: Semana 6, 24/09
  - Control 2: Semana 11, 05/11
  - Control 3: Semana 15, 03/12
  - Control Recuperativo: Semana 16, 10/12

Evaluaciones

- Controles (75%, potenciales fechas de controles):
  - Control 1: Semana 6, 24/09
  - Control 2: Semana 11, 05/11
  - Control 3: Semana 15, 03/12
  - Control Recuperativo: Semana 16, 10/12
- Tareas (25%, potenciales fechas de entrega):
  - Tarea 1: Semana del 15 Septiembre (15/09).
  - Tarea 2: Semana del 27 Octubre (27/10).
  - Tarea 3: Semana del 24 Noviembre (24/11).

Condiciones y Políticas de Evaluación

- Se evaluará el aprendizaje del contenido presentado en las cátedras y en las ayudantías, mediante tres actividades complementarias (tareas, ejercicios) y tres controles de cátedra. Las ponderaciones de cada instancia de evaluación son las siguientes:
    - 1. Calificaciones en actividades complementarias 25%.
    - 2. Calificaciones en controles de cátedra 75%.
- La Nota Final del curso se calculará considerando las ponderaciones anteriores
- La aprobación de la asignatura está sujeta a las condiciones Nota Cátedra ≥ 4.0 y Nota de Actividades Complementarias ≥ 4.0.

Condiciones y Políticas de Evaluación

- Estudiantes que se ausenten a un control tendrán la oportunidad de recuperarlo con el examen. La nota del examen reemplazará la nota más baja de los controles de la asignatura, solo en caso de ser la nota de examen superior.
- Un/a estudiante que cometa plagio sobtendrá un 1,0 en la evaluación y el caso será informado a Escuela de Ingeniería.

Materiales

- Repositorio GitHub – `https://github.com/adigenova/uohpmd`
  - Slides Clases (Lunes [M02] y Miércoles [?] : 12:00 - 13:30)
  - Código
- Ayudante : X Y (Viernes 14:30 - 16:00, A305)

Materiales

- Repositorio GitHub – `https://github.com/adigenova/uohpmd`
  - Slides Clases (Lunes [M02] y Miércoles [?] : 12:00 - 13:30)
  - Código
- Ayudante : X Y (Viernes 14:30 - 16:00, A305)
- Ucampus – `https://ucampus.uoh.cl/uoh/2025/2/COM4002/`
  - Comunicación (Consultas, noticias, evaluaciones)
  - Planificación

Materiales

- Repositorio GitHub – `https://github.com/adigenova/uohpmd`
  - Slides Clases (Lunes [M02] y Miércoles [?] : 12:00 - 13:30)
  - Código
- Ayudante : X Y (Viernes 14:30 - 16:00, A305)
- Ucampus – `https://ucampus.uoh.cl/uoh/2025/2/COM4002/`
  - Comunicación (Consultas, noticias, evaluaciones)
  - Planificación
- Bibliografía
  - S. Tanenbaum, M. Van Steen. Distributed Systems: Principles and Paradigms (2nd Edition). Prentice Hall, 2006
  - P. J. Sadalage, M. Fowler. NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence. Addison-Wesley Professional, 2012

Resultados de Aprendizaje

- Conocer los principios fundamentales de diseño de sistemas distribuidos y paralelos (Linux)

Resultados de Aprendizaje

- Conocer los principios fundamentales de diseño de sistemas distribuidos y paralelos (Linux)
- Implementar comunicación entre procesos de una máquina (OpenMP, pthread)

Resultados de Aprendizaje

- Conocer los principios fundamentales de diseño de sistemas distribuidos y paralelos (Linux)
- Implementar comunicación entre procesos de una máquina (OpenMP, pthread)
- Implementar comunicación entre máquinas (MPI)

Resultados de Aprendizaje

- Conocer los principios fundamentales de diseño de sistemas distribuidos y paralelos (Linux)
- Implementar comunicación entre procesos de una máquina (OpenMP, pthread)
- Implementar comunicación entre máquinas (MPI)
- Utilizar Nextflow/Hadoop para distribuir tareas computacionales en un clúster.

Resultados de Aprendizaje

- Conocer los principios fundamentales de diseño de sistemas distribuidos y paralelos (Linux)
- Implementar comunicación entre procesos de una máquina (OpenMP, pthread)
- Implementar comunicación entre máquinas (MPI)
- Utilizar Nextflow/Hadoop para distribuir tareas computacionales en un clúster.
- Conocer la taxonomía de modelos de datos NoSQL, sus lenguajes de consulta y manipulación de datos con modelos NO-SQL.

Resultados de Aprendizaje

- Conocer los principios fundamentales de diseño de sistemas distribuidos y paralelos (Linux)
- Implementar comunicación entre procesos de una máquina (OpenMP, pthread)
- Implementar comunicación entre máquinas (MPI)
- Utilizar Nextflow/Hadoop para distribuir tareas computacionales en un clúster.
- Conocer la taxonomía de modelos de datos NoSQL, sus lenguajes de consulta y manipulación de datos con modelos NO-SQL.
- Construir y manipular una base de datos distribuida.

Consultas?

Consultas o comentarios?

Muchas Gracias.