Actas de las II Jornadas Uruguayas de Ciencias de la Computación 2025 “Ida Holz”

Instituto de Computación, Facultad de Ingeniería
Universidad de la República, Uruguay

Editores

Sergio Nesmachnow
Pedro Moreno
Diego Rossit

nil novum sub sole

UNIVERSIDAD DE LA REPÚBLICA URUGUAY

CES CENTRO DE ENSAYOS DE SOFTWARE

Editores

Sergio Nesmachnow
Universidad de la República, Uruguay

Pedro Moreno
Universidad Autónoma del Estado de Morelos, México

Diego Rossit
Universidad Nacional del Sur, Argentina

Elaborado por el Comité Organizador de las Jornadas Uruguayas de Ciencias de la Computación.

Cómo citar este libro: Nesmachnow, S., Moreno, P. y Rossit, D. (eds). (2025). Actas de las II Jornadas Uruguayas de Ciencias de la Computación. Serie Reportes Técnicos InCo-PEDECIBA.
ISSN 3046–4277.

Prefacio

Este volumen presenta artículos seleccionados de las II Jornadas Uruguayas de Ciencias de la Computación “Ida Holz”, celebradas del 8 al 12 de diciembre de 2025 en la Facultad de Ingeniería, Universidad de la República, Uruguay, en modalidad presencial, con charlas temáticas y presentaciones científicas.

Los objetivos principales de las Jornadas Uruguayas de Ciencias de la Computación incluyeron la diseminación, divulgación y comunicación entre la academia (investigadores, docentes), profesionales, empresas y estudiantes de grado y posgrado en temas relacionados con la informática. Las jornadas proporcionaron un foro para que investigadores, científicos, profesores, empresarios, tomadores de decisiones, estudiantes de posgrado y profesionales de Uruguay y otros países compartieran sus iniciativas actuales relacionadas con diversos ámbitos de la informática. Se desarrollaron bajo un enfoque que combinó temas y actividades académicas, profesionales y empresariales.

El programa principal consistió en cinco workshops temáticos que incluyeron catorce charlas de expertos internacionales, dos sesiones de empresas e industria, una sesión de presentaciones de artículos científicos, un seminario de estudiantes de posgrado y cuatro talleres para estudiantes y público con conocimientos básicos e intermedios de informática. Los workshops correspondieron a las temáticas de Grandes modelos de lenguaje, Robótica autónoma, Ontologías y sus aplicaciones, Procesamiento de archivos del pasado reciente y Ciudades inteligentes integradas, eficientes y sostenibles. En conjunto, se celebró la ceremonia de graduación de los programas profesionales y de educación permanente del Centro de Ensayos de Software. En total, más de 500 personas participaron en las diferentes actividades de las jornadas.

El Comité del Programa recibió 16 artículos científicos para su evaluación y presentación en las jornadas. Todos los trabajos fueron seleccionados para ser publicados en este libro. Los artículos fueron sometidos a un proceso de revisión por pares por al menos tres expertos antes de ser seleccionados para su publicación.

Expresamos nuestro profundo agradecimiento a todos los colaboradores de las Jornadas Uruguayas de Ciencias de la Computación, a los integrantes de los comités académicos y de organización y a los autores y revisores por su esfuerzo, que hizo que el proceso de revisión de los artículos fuera eficiente. También agradecemos a los participantes de las Jornadas Uruguayas de Ciencias de la Computación, a todas las instituciones participantes y colaboradoras, a nuestras instituciones y a todos los lectores de este volumen.

Diciembre de 2025

[Sergio Nesmachnow, Pedro Moreno, Diego Rossit]

UNIVERSIDAD
DE LA REPÚBLICA
URUGUAY

CES CENTRO DE
ENSAYOS
DE SOFTWARE

Organización

Comité organizador

Javier Baliosian

Gonzalo Tejera

Sergio Nesmachnow

Libertad Tansini

Guillermo Moncecchi

Comité Académico

Javier Baliosian

Gonzalo Tejera

Sergio Nesmachnow

Libertad Tansini

Guillermo Moncecchi

Adriana Marotta

Aiala Rosá

Alberto Pardo

Andrea Delgado

Antonio Mauttone

Carlos Luna

Carlos Torres

Diego Rossit

Dina Wonsever

Eduardo Redondo

Luis Chiruzzo

Marcos Viera

Facundo Benavides

Eduardo Fernández

Flavia Serra

Guillermo Trinidad

Gustavo Betarte

Guzmán Llambías

Juan García-Garland

Laura González

Lorena Etcheverry

Pedro Moreno

Pedro Piñeyro

Raquel Sosa

Juan Prada

Renzo Massobrio

Sebastián Pizard

Yamila Grassi

Contenido

Genetic Ancestry Detection using HyenaDNA: The Critical Role of Positional Embeddings

Gonzalo Cameto[1], María Inés Fariello[1,2], Federico Lecumberry[1]

1- Facultad de Ingeniería, Universidad de la República, Montevideo, Uruguay
2 - Unidad de Bioinformática, Instituto Pasteur de Montevideo
gonzalo.cameto@fing.edu.uy

Abstract. The human genome contains 3.2 billion nucleotides with dependencies spanning over 100,000 positions, yet Transformer-based models are limited to contexts below 4,000 tokens due to quadratic $O(L^2)$ complexity. HyenaDNA addresses this limitation through a subquadratic $O(L \log L)$ architecture combining implicit convolutions with multiplicative gating. We demonstrate that HyenaDNA is highly effective for genetic ancestry detection, achieving up to 99.89% accuracy on continental classification using Single Nucleotide Polymorphisms (SNPs) from the 1000 Genomes Project, with strong performance across context lengths (96.64%–99.34% for 1,024–4,096 SNPs). Our key finding reveals that Factorized Positional Embeddings (FPE) are not merely beneficial but *essential*, representing 97.3% of model parameters and causing performance to collapse to near-random levels (34.6% accuracy) when removed. We analyze the critical trade-off controlled by the position scaling factor α, which groups consecutive genomic positions: decreasing α improves performance but dramatically increases memory consumption, with $\alpha = 100$ providing a good balance. Comparative experiments using sequences of 1,024 SNPs, demonstrate HyenaDNA's advantages over Transformers: 3.75× less GPU memory, 2.56× faster training, and +4% higher accuracy at equivalent parameter counts. These results establish that subquadratic architectures combined with rich positional encoding and two-stage training constitute an effective and accessible approach for genomic sequence analysis.

Keywords: HyenaDNA · Genetic ancestry · SNPs · Factorized positional embeddings · Subquadratic architectures

1 Introduction

The human genome's 3.2 billion nucleotides with dependencies exceeding 100,000 base pairs (bp) pose fundamental challenges for deep learning. Transformer-based models [10] exhibit quadratic $O(L^2)$ complexity, limiting genomic applications to <4,000 tokens [4].

SNPs—occurring every 600-700 bp—preserve evolutionary history crucial for ancestry detection [1, 7]. However, these sparse signals require positional context to capture linkage patterns across extended genomic regions.

HyenaDNA [6] achieves subquadratic $O(L \log L)$ complexity via the Hyena operator [8], combining implicit convolutions (evaluated via Fast Fourier Transform, FFT) with multiplicative gating, enabling 1M-token contexts at single-nucleotide resolution.

We demonstrate HyenaDNA's effectiveness for ancestry detection using 1000 Genomes SNP data. Our key finding: **Positional Embeddings are essential**, with FPE preferred for its reduced memory footprint—representing 97.3% of parameters yet causing accuracy to collapse to near-random levels (34.6%) when removed. The scaling factor α controls memory-performance trade-off, with $\alpha = 100$ offering a good balance for consumer GPUs. We achieve accuracy ranging from 96.64% (1,024 SNPs) to 99.89% (32,768 SNPs), with 3.75× less memory and 2.56× faster training than Transformers (compared using 1,024 SNPs).

2 Background

2.1 The Hyena Operator and HyenaDNA

The Hyena operator [8] replaces quadratic attention with subquadratic primitives. For input $\mathbf{u} \in \mathbb{R}^L$, the operator factorizes as:

$$\mathbf{y} = \mathbf{D}_{\mathbf{x}}^N \mathbf{S}_{\mathbf{h}}^N \cdots \mathbf{D}_{\mathbf{x}}^2 \mathbf{S}_{\mathbf{h}}^2 \mathbf{D}_{\mathbf{x}}^1 \mathbf{S}_{\mathbf{h}}^1 \mathbf{v}\,, \tag{1}$$

where Toeplitz matrices $\mathbf{S}_{\mathbf{h}}^n$ implement causal convolutions and diagonal matrices $\mathbf{D}_{\mathbf{x}}^n$ provide multiplicative gating. Filters are generated implicitly via Multi-Layer Perceptrons (MLPs) ($h_t^n = f_\theta(t)$), decoupling parameters from length. Convolutions evaluate in $O(L \log L)$ via FFT: $\mathbf{y} = \mathrm{FFT}^{-1}(\mathrm{FFT}(\mathbf{h}) \odot \mathrm{FFT}(\mathbf{x}))$.

Unlike Transformers, where attention is permutation-invariant and requires explicit positional encodings to capture sequence order, Hyena's causal convolutions inherently encode ordering through their sequential structure—each output depends only on preceding inputs in a fixed positional relationship.

HyenaDNA [6] applies this to genomic sequences using character-level tokenization (A, C, G, T representing adenine, cytosine, guanine, thymine, plus N for unknown bases), preserving single-nucleotide resolution. Pre-trained models support contexts from 1k to 1M tokens.

2.2 Positional Encoding for Genomics

While Hyena's convolutions capture relative ordering within the input sequence, working with SNPs introduces a distinct challenge: SNPs are sparse markers extracted from specific genomic coordinates, not consecutive nucleotides. When we form a sequence of SNP alleles, the original genomic distances between them are lost—two adjacent SNPs in our input may be thousands of base pairs apart in the genome. This loss of absolute positional information makes explicit positional encodings essential for SNP-based tasks.

In genomics, absolute coordinates carry biological significance—proximity to regulatory elements, distance between linked variants, and chromosomal context all influence functional interpretation.

Standard Approaches. The original Transformer [10] uses sinusoidal Positional Encoding (PE):

$$\text{PE}_{(pos,2i)} = \sin(pos/10000^{2i/d}), \quad \text{PE}_{(pos,2i+1)} = \cos(pos/10000^{2i/d}), \tag{2}$$

where pos is position and d is embedding dimension. Alternatively, learned positional embeddings use a parameter matrix $\mathbf{E} \in \mathbb{R}^{P \times d}$ where row $\mathbf{E}_{pos}$ provides the positional vector for position pos.

The Genomic Challenge. For a human chromosome with $P \approx 2.5 \times 10^8$ positions and $d = 128$, a full learned embedding matrix requires:

$$\text{Memory} = P \times d \times \text{bytes/param} = 2.5 \times 10^8 \times 128 \times 2 \approx 60 \text{ GB}, \tag{3}$$

assuming float16 precision for parameters alone; optimizers like AdamW store additional states (momentum, variance), further multiplying memory requirements. This is prohibitive for whole-chromosome or whole-genome analysis.

Factorized Positional Embeddings. FPE addresses this via low-rank factorization:

$$\mathbf{E} = \mathbf{AB}, \tag{4}$$

where $\mathbf{A} \in \mathbb{R}^{P \times k}$ and $\mathbf{B} \in \mathbb{R}^{k \times d}$ with factorization rank $k \ll d$. This reduces memory complexity from $O(Pd)$ to $O(Pk + kd)$. For $k = 16$:

$$\text{Memory}_{\text{FPE}} = (P \times k + k \times d) \times 2 = (2.5 \times 10^8 \times 16 + 16 \times 128) \times 2 \approx 7.5 \text{ GB}, \tag{5}$$

achieving an 8× reduction while maintaining expressiveness through learnable low-rank structure.

Scaling Factor α. To further reduce memory, a scaling factor α groups consecutive positions:

$$P_{\text{effective}} = \left\lfloor \frac{P_{\text{chromosome}}}{\alpha} \right\rfloor. \tag{6}$$

All positions within a bin of size α share the same embedding. This trades positional resolution for memory: larger α reduces parameters but loses fine-grained discrimination. For SNP-based tasks where positions are genomic coordinates (potentially millions of base pairs apart), appropriate α balances positional resolution and memory consumption. As demonstrated in our experiments (Section 4.1), α controls a critical performance-memory trade-off essential for accessible hardware deployment.

2.3 Related Work

Genomic Models. DNABERT [4] uses k-mer tokenization, sacrificing single-nucleotide resolution. Enformer [2], a Convolutional Neural Network (CNN) combined with Transformer, achieves ∼100 kilobases (kb) contexts via downsampling, losing variant-level information. Both face quadratic scaling limits.

Ancestry Methods. Gnomix [3] achieves 98.92% accuracy distinguishing 8 populations using chromosome-specific XGBoost models with ∼4,587 SNP

windows. Our model classifies 3 continental populations (European (EUR), East Asian (EAS), African (AFR)) using a single unified architecture with comparable context length.

Positional Encodings. Rotary Position Embedding (RoPE) [9] and Fourier methods avoid learnable parameters. We show (Section 4.1) that learnable FPE is essential for SNP tasks.

Table 1 summarizes how our approach compares to existing genomic foundation models. HyenaDNA's subquadratic scaling enables 10–1000× longer contexts than Transformer-based models while maintaining single-nucleotide resolution.

Table 1. Comparison of genomic models for ancestry and sequence analysis

Model	Architecture	Complexity	Max Context	Resolution
DNABERT [4]	Transformer	$O(L^2)$	512 tokens	k-mer (6bp)
Enformer [2]	CNN + Transf.	$O(L^2)$	∼200 kb	Binned (128bp)
Gnomix [3]	XGBoost	$O(N \log N)$	∼4,587 SNPs	Single SNP
HyenaDNA	Hyena	$O(L \log L)$	1M bp	Single nucleotide

Note that for local ancestry inference, relatively small windows (128–4,096 SNPs) are sufficient, as ancestry signals are well-captured within these ranges. Both Gnomix and our approach operate within similar context lengths for this task. HyenaDNA's extended capacity (up to 1M bp) enables other genomic applications requiring longer-range dependencies, such as regulatory element detection or whole-chromosome analysis.

3 Methodology

3.1 Dataset and Preprocessing

We use 1000 Genomes data [1]: 590 individuals from European (EUR, 240), African (AFR, 160), and East Asian (EAS, 190) populations. Variant Call Format (VCF) files were processed to extract ∼4.5M SNPs per individual, stored in Hierarchical Data Format 5 (HDF5). Stratified sampling: (1) select population, (2) individual, (3) chromosome, (4) random position, (5) extract window (128-32,768 SNPs). Data splits at the individual level: 70% train, 15% validation, 15% test, ensuring no individual appears in multiple sets. From each split, we generated 10k training sequences, 5k validation, and 5k test sequences, with fixed seeds for reproducibility.

3.2 Model Architecture

Table 2 shows parameter distribution: FPE dominates with 97.3% (39.8M), HyenaDNA encoder 2.7% (1.1M), totaling 40.9M parameters.

Table 2. Parameter distribution

Component	Params	%
HyenaDNA Encoder	1.1M	2.7
FPE (α=100)	39.8M	97.3
Total	**40.9M**	**100**

Components: SNPs tokenized as alleles (A,C,G,T). FPE: $\mathbf{E} = \mathbf{AB}$ where $\mathbf{A} \in \mathbb{R}^{P_{\text{eff}} \times k}$, $\mathbf{B} \in \mathbb{R}^{k \times d}$ ($k = 16$, $d = 128$, $P_{\text{eff}} = \lceil P_{\text{chr}}/\alpha \rceil$). 4-layer HyenaDNA encoder ($d = 128$) processes combined token and positional embeddings. MLP classification head operates on the last token's hidden state to output ancestry probabilities.

3.3 Two-Stage Training Strategy

Stage 1: Next-Token Prediction. The model predicts the next SNP s_{t+1} given the context $(s_1, \ldots, s_t)$ by minimizing the cross-entropy loss defined as:

$$\mathcal{L}_{\text{LM}} = -\frac{1}{L} \sum_{t=1}^{L} \log P(s_{t+1} | s_1, \ldots, s_t) \tag{7}$$

This allows the model to learn linkage patterns in a self-supervised manner. We train for up to 100 epochs employing early stopping based on validation perplexity.

Stage 2: Classification. The pre-trained model is fine-tuned for ancestry inference using the classification loss $\mathcal{L}_{\text{cls}} = -\sum_{c=1}^{C} y_c \log \hat{y}_c$, where C represents the number of ancestry classes. *Hyperparameters:* AdamW optimizer, learning rate $= 6 \times 10^{-5}$, trained up to 300 epochs with early stopping (patience = 50). *Hardware:* NVIDIA RTX 4060 Ti (16GB), demonstrating accessibility via the subquadratic efficiency of the architecture.

4 Experiments and Results

4.1 The Critical Role of FPE and Impact of α

Table 3 compares positional encodings (1024 SNPs, two-stage training).

Results confirm that learnable positional encoding is essential for next-token prediction on SNP sequences. Without positional information, perplexity remains at 4.00 (random baseline). Fixed encodings (sinusoidal, RoPE, Fourier) provide minimal improvement (3.83–4.00), while FPE achieves substantially lower perplexity. The scaling factor α controls a memory-performance trade-off: reducing α improves perplexity but increases memory consumption. FPE with α=100 offers an effective balance (perplexity 1.72, 0.6 GB), achieving 97% classification accuracy after fine-tuning (Stage 2).

Table 3. Positional encoding comparison

Encoding	Perp.	Mem (GB)
No position	4.00	–
Sinusoidal/RoPE/Fourier	3.83-4.00	–
FPE (α=1000)	2.79	0.06
FPE (α=100)	**1.72**	0.6
FPE (α=10)	1.41	5.96

4.2 Context Length Scaling

We systematically evaluated model performance across different context lengths, measuring not only accuracy but also computational requirements and model quality metrics. Table 4 presents comprehensive results for contexts ranging from 128 to 32,768 SNPs.

Table 4. Context length scaling: comprehensive metrics. t/ep = time per epoch (mm:ss), Total = total training time (Next-Token Prediction (NTP) + Classification), Mem = GPU memory (GB), Batch = batch size used.

Context	α	Batch	Acc	Perp	t/ep NTP	t/ep Cls	Total	Mem
128	100	256	75.11%	2.12	0:08	0:09	1.2h	2.9
128	10	256	83.24%	1.72	0:24	0:27	3.2h	11.0
512	100	256	92.07%	1.84	0:18	0:18	2.5h	7.3
1,024	100	256	96.64%	1.86	0:40	0:31	3.5h	13.1
2,048	100	128	98.38%	1.68	1:06	1:01	8.1h	12.9
4,096	100	64	99.34%	1.66	1:51	1:51	12.3h	12.5
32,768	100	6	**99.89%**	**1.62**	28:00	22:00	23.6h	11.5

Accuracy improves substantially with context length: from 75% at 128 SNPs to 97% at 1,024 SNPs (+21.5 pp), continuing to 99.89% at 32k SNPs as the model leverages longer-range linkage patterns. Lower perplexity in Stage 1 correlates with higher classification accuracy in Stage 2, validating the two-stage design. Computational costs scale subquadratically—NTP time grows only 210× for 256× more data, and memory remains bounded (11–13 GB for 1k–32k SNPs) by adjusting batch size, enabling long contexts on consumer hardware. At short contexts, α significantly impacts performance: reducing it from 100 to 10 at 128 SNPs improves accuracy by 8% at the cost of 4× higher memory.

4.3 HyenaDNA vs Transformer

Controlled comparison: same ∼41M parameters, FPE(α=100), two-stage training; only difference: Hyena operator vs multi-head attention (Table 5).

Table 5. HyenaDNA vs Transformer (1024 SNPs)

Metric	Trans.	Hyena	Impr.
GPU Mem (MB)	14,786	3,940	**3.75**×
Time/epoch	1:40	0:39	**2.56**×
Perplexity	2.78	2.38	-14%
Accuracy	86%	90%	+**4%**

HyenaDNA's $O(L \log L)$ vs Transformer's $O(L^2)$ yields dramatic efficiency gains.

4.4 Latent Space Analysis

t-distributed Stochastic Neighbor Embedding (t-SNE) visualization [5] reveals representation transformation across training stages (Figures 1, 2).

Fig. 1. t-SNE after Stage 1: 4-pointed star organized by next nucleotide (A,C,G,T), showing learned linkage patterns.

Stage 1 learns nucleotide prediction (star geometry); Stage 2 reorganizes to population structure (separated clusters). Misclassifications concentrate at cluster boundaries. This demonstrates successful transfer learning from self-supervised to supervised tasks.

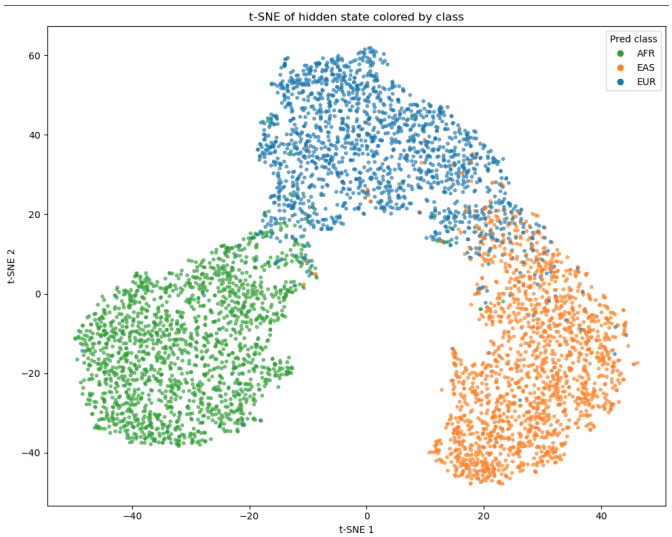


Fig. 2. t-SNE after Stage 2: Complete reorganization into 3 population clusters (AFR, EAS, EUR) with EUR geometrically between AFR-EAS.

4.5 Model Performance Analysis

Different context lengths offer distinct advantages for genomic applications. Short windows (128–1024 SNPs) enable fine-grained local ancestry inference and sliding-window analysis across chromosomes, while contexts of 2048–4096 SNPs provide an optimal balance between accuracy and computational cost for this task. The 32,768-SNP experiment was conducted primarily to evaluate the model's capabilities at extended context lengths, demonstrating that HyenaDNA can effectively leverage longer-range dependencies when available. Table 6 shows the confusion matrix for 4096 SNPs, achieving **99.34% accuracy** with F1=0.99 per class. The maximum accuracy observed was **99.89%** at 32768 SNPs (Table 4).

Table 6. Confusion matrix (4096 SNPs)

True/Pred	EUR	AFR	EAS
EUR	**0.993**	0.004	0.003
AFR	0.006	**0.991**	0.003
EAS	0.005	0.001	**0.994**

Gnomix [3] achieves 98.92% accuracy using 22 chromosome-specific XGBoost models trained on ∼4,587 SNPs. Our unified end-to-end model matches or ex-

ceeds this performance: 99.34% at 4,096 SNPs and 99.89% at 32,768 SNPs. Even shorter contexts suitable for sliding-window analysis achieve strong results (96.64% at 1,024 SNPs). Training requires 3.5–23.6h depending on context length (RTX 4060 Ti). Inference is highly efficient: a complete genome (∼1,600 sliding windows across 22 chromosomes) processes in ∼15 seconds at ∼106 windows/s.

4.6 Chromosome-Level Ancestry Visualization

To demonstrate the model's ability to capture fine-grained ancestry patterns, we visualize predictions along all 22 autosomal chromosomes for representative individuals from each population. Figure 3 shows ancestry probability maps obtained by sliding a 1,024-SNP window across each chromosome and displaying the predicted class.

Interpretation. The ancestry maps reveal the model's ability to consistently identify the correct ancestry class across the entire genome. For individuals from reference populations, the predicted ancestry remains predominantly uniform across all 22 chromosomes, with the expected class dominating the vast majority of genomic windows. While occasional regions may show different predictions—which could reflect either model uncertainty or genuine ancestral complexity even in "reference" individuals—the overall pattern demonstrates that the model successfully captures genome-wide ancestry signals rather than relying on isolated genomic regions.

These visualizations confirm that the high classification accuracy observed in aggregate metrics reflects consistent predictions across the genome, not just correct classification of a few highly informative regions. This genome-wide consistency suggests the model has learned to leverage distributed ancestry-informative patterns throughout the chromosomes.

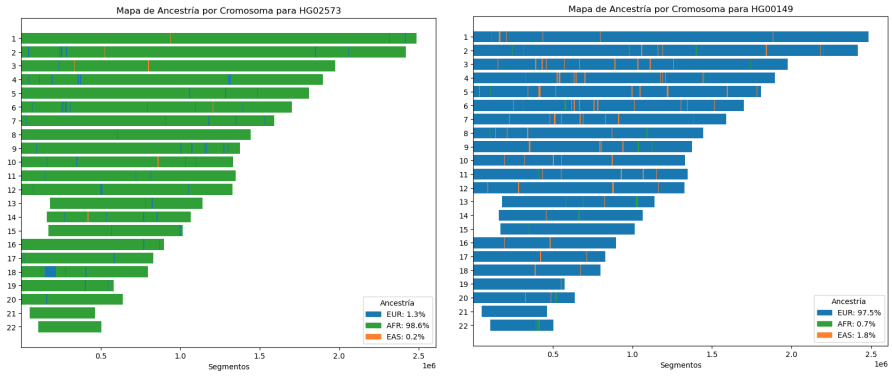
5 Discussion

5.1 Principal Finding: FPE Essentiality

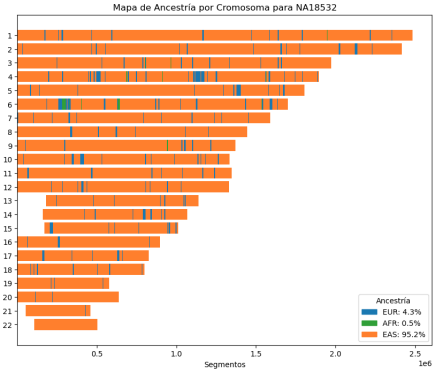
Our most significant finding is the *critical role* of FPE for SNP-based genomic tasks. FPE represents 97.3% of model parameters (39.8M of 40.9M), yet ablation experiments (Table 3) demonstrate that removing this component causes performance to collapse—perplexity remains at the random baseline (4.00) and classification accuracy drops to 34.6% (near the 33.3% random baseline for 3 classes).

Why is positional information so critical? SNPs are sparse markers filtered from the full genome, occurring approximately once every 600-700 base pairs. Without positional information, the model sees only unordered collections of alleles (A, C, G, T), losing crucial biological context:

1. **Linkage disequilibrium (LD)**: Non-random associations between alleles at different loci, which vary characteristically by population due to demographic history, bottlenecks, and admixture events



(a) African (Gambian) **(b)** European (British)



(c) East Asian (Japanese)

Fig. 3. Ancestry probability maps across all 22 autosomal chromosomes for representative individuals. (a) African individual showing dominant AFR ancestry (green). (b) European individual showing consistent high EUR probability (blue). (c) East Asian individual showing clear EAS ancestry (orange). Each position represents a sliding window of 1,024 SNPs. Color indicates predicted ancestry class. The model maintains consistent predictions across all chromosomes, with minimal fluctuations.

2. **Haplotype structure**: Combinations of alleles inherited together on chromosome segments, reflecting population-specific recombination histories
3. **Recombination landscapes**: Genomic regions with high/low recombination rates shape genetic diversity differently across populations and create population-specific patterns

Fixed positional encodings (sinusoidal PE, RoPE) provide only marginal improvement (perplexity 3.83–4.00, near random baseline), as they cannot adaptively learn which genomic positions are most informative for ancestry discrimination. Only rich learnable embeddings like FPE, with sufficient parameter capacity and appropriate granularity (α=100), enable the model to discover and exploit position-specific ancestry signals embedded in linkage patterns.

Implications. This finding challenges the common practice in genomic deep learning of treating positional encoding as a secondary architectural detail. For SNP-based tasks, positional embeddings should be viewed as a first-class component requiring careful design and substantial parameter budget.

5.2 The α Trade-off and Practical Deployment

The scaling factor α in FPE controls a trade-off between positional resolution and memory consumption. As shown in Table 3, reducing α from 1000 to 10 decreases perplexity from 2.79 to 1.41, but increases FPE memory from 0.06 to 5.96 GB (100×). For continental ancestry detection, α=100 provides a good balance (perplexity 1.72, 0.6 GB, 96.64% classification accuracy at 1,024 SNPs), fitting within consumer GPU budgets. The optimal α is problem-dependent: tasks with longer-range dependencies may tolerate coarser resolution, while fine-grained local ancestry inference may benefit from smaller values.

5.3 Subquadratic Architecture Advantages

Our controlled comparison demonstrates clear advantages of HyenaDNA's $O(L \log L)$ complexity over Transformer's $O(L^2)$:

- **Memory efficiency**: 3.75× reduction at 1024 SNPs enables longer contexts and larger batches
- **Training speed**: 2.56× faster per epoch accelerates experimentation
- **Model quality**: +4% accuracy improvement even at modest contexts

Critically, these advantages manifest even at relatively short sequences (1024 tokens) where both architectures theoretically fit in memory. To characterize maximum sequence capacity, we evaluated both architectures with batch size 1 on 16GB GPU memory: the Transformer reached its limit at approximately 8,500 SNPs, while HyenaDNA scaled to approximately 210,000 SNPs—a 24× advantage in maximum context length. This demonstrates that for genomic applications requiring long-range context—capturing regulatory interactions spanning megabases, phasing haplotypes, analyzing whole chromosomes—subquadratic architectures are not merely more efficient but often necessary.

5.4 Two-Stage Training and Transfer Learning

The two-stage protocol exemplifies effective domain-specific transfer learning. Stage 1 (next-token prediction) learns general patterns of linkage and haplotype structure in a self-supervised manner, requiring no ancestry labels. The t-SNE visualization (Figure 1) shows nucleotide-organized representations—a four-pointed star where each arm corresponds to contexts favoring A, C, G, or T.

Stage 2 (supervised classification) repurposes these learned representations for ancestry prediction. The t-SNE visualization (Figure 2) shows complete reorganization into three population-specific clusters with minimal overlap. This transformation demonstrates that linkage patterns learned in Stage 1 provide a strong inductive bias for population discrimination, analogous to how language models pre-trained on next-word prediction transfer to downstream Natural Language Processing (NLP) tasks.

5.5 Biological Interpretation and Model Behavior

The model learns continental population structure using only SNP sequences and ancestry labels—no explicit geographic or historical information. Several observations suggest biologically meaningful learned representations:

1. **Clear separation**: Three continental groups form distinct, well-separated clusters consistent with known genetic differentiation
2. **Geographic plausibility**: Europe's intermediate position between Africa and East Asia (Figure 2) mirrors physical geography and historical migration routes
3. **Uncertainty patterns**: Misclassifications concentrate at cluster boundaries, likely reflecting individuals with admixed ancestry or from populations near continental transitions
4. **Chromosome-level consistency**: Ancestry maps (Figure 3) show stable predictions across entire chromosomes for homogeneous individuals, validating robustness

While our evaluation focuses on computational performance, these patterns suggest the model discovers genuine population genetic structure rather than spurious correlations.

6 Limitations and Future Work

6.1 Current Limitations

Population Scope. Our evaluation is limited to three broad continental groups (Africa, Europe, East Asia). Real-world ancestry inference applications require significantly finer resolution—distinguishing between subpopulations within continents (e.g., Northern vs Southern European, East vs West African), quantifying admixture proportions for individuals with complex ancestry, and extending

coverage to additional continental groups (Americas, Oceania, South Asia, Middle East). The 1000 Genomes populations we selected represent reference panels with relatively homogeneous ancestry; extension to 10-20 populations or admixed individuals would better reflect practical deployment scenarios.

FPE Memory Footprint. Despite the factorization strategy, FPE still dominates model parameters (97.3%, 39.8M of 40.9M). While the scaling factor α provides a tunable memory-performance trade-off, the steep scaling (100× memory increase from α=1000 to α=10) limits practical exploration of very fine positional resolution. More parameter-efficient positional encoding methods that maintain FPE's expressive power while dramatically reducing memory consumption would enable broader accessibility and exploration of extreme context lengths.

Transformer Comparison Scope. Our controlled comparison used 1,024-SNP contexts for practical training with reasonable batch sizes. Maximum sequence length experiments (batch size 1, 16GB GPU) revealed the Transformer can scale to approximately 8,500 SNPs before exhausting memory, while HyenaDNA reaches approximately 210,000 SNPs—a 24× advantage. However, at these extreme lengths training becomes impractical (batch size 1), and we did not conduct full training runs beyond our comparison point. Future work could explore performance at intermediate contexts (4,096–8,000 SNPs) where both architectures remain feasible but HyenaDNA's efficiency advantages become more pronounced.

Context Range Explored. We tested contexts up to 32,768 SNPs, achieving perplexity of 1.62 after 39 epochs (28 min/epoch, batch size 6). HyenaDNA's architecture supports up to 1 million tokens, suggesting potential for even longer contexts—64k SNPs (100-200 megabases of genome), 350k SNPs (whole chromosome), or multi-chromosome analysis. Exploring these extreme context lengths could reveal new applications (e.g., detecting structural variants, capturing ultra-long-range regulatory interactions) but requires substantial computational resources.

6.2 Future Research Directions

Alternative Positional Encodings. While FPE is essential for performance, the current implementation is suboptimal: SNPs appear approximately every 600–700 base pairs, meaning only a small fraction of encoded positions are ever used. Setting α=1 increases resolution but allocates parameters for many positions that will never contain a SNP. Developing encodings that achieve high resolution while exploiting this sparsity—through hierarchical factorization, learnable compression, or hybrid fixed/learnable approaches—represents an important research direction.

Extended Contexts. Investigating performance with 64k-1M SNP contexts could enable novel applications: (1) *Whole-chromosome processing*—analyzing entire chromosomes (50k-350k SNPs) without sliding windows, capturing global chromosome-level patterns; (2) *Long-range interactions*—detecting regulatory dependencies spanning megabases, relevant for enhancer-promoter interactions

and topologically associating domains; (3) *Structural variation*—identifying large-scale genomic rearrangements, copy number variations, and inversions through discontinuities in ancestry patterns.

Mixed Ancestry and Admixture. Extending beyond discrete classification to quantitative admixture estimation (e.g., "60% European, 30% African, 10% Native American") would address real-world scenarios where individuals have complex multi-way ancestry. This could leverage the chromosome-level visualization framework (Figure 3) to identify admixture breakpoints—genomic positions where ancestry switches due to historical recombination events. Such fine-grained ancestry inference is valuable for understanding population history and correcting for population stratification in genetic association studies.

7 Conclusions

HyenaDNA achieves up to 99.89% accuracy for continental ancestry detection (1000 Genomes SNPs), with strong performance across context lengths—from 96.64% (1,024 SNPs) to 99.34% (4,096 SNPs)—demonstrating subquadratic architectures' effectiveness for genomic tasks at multiple scales.

Key Findings: (1) **FPE essential**—97.3% of parameters; without it, perplexity remains at random baseline (4.00) and classification accuracy collapses to 34.6%; positional encoding is fundamental for sparse genomic variants. (2) α **trade-off characterized**—α=100 provides optimal balance (96.64% accuracy at 1,024 SNPs, 0.6 GB memory); decreasing α improves performance but exponentially increases memory. (3) **Subquadratic advantages demonstrated**—3.75× less memory, 2.56× faster, +4% accuracy vs Transformers at 1,024 SNPs; 24× longer maximum context (210k vs 8.5k SNPs). (4) **Two-stage training effective**—self-supervised next-token prediction transfers successfully to supervised classification.

Implications: Subquadratic architectures combined with rich positional encoding constitute an effective approach for genomic analysis. The critical role of FPE suggests that developing more parameter-efficient positional encoding alternatives should be a research priority.

Acknowledgments. This research was conducted as part of the Master's program in Mathematical Engineering at Universidad de la República, Uruguay. The author gratefully acknowledges the guidance of thesis advisors Prof. María Inés Fariello, Prof. Federico Lecumberry, and Prof. Marcelo Fiori. Computational experiments were performed on consumer-grade hardware, demonstrating the accessibility enabled by efficient architectures.

Disclosure of Interests. The author declares no competing interests relevant to the content of this article. All data used in this study are from publicly available sources (1000 Genomes Project) with appropriate consent for research use.

References

1. 1000 Genomes Project Consortium: A global reference for human genetic variation. Nature **526**(7571), 68–74 (2015). https://doi.org/10.1038/nature15393
2. Avsec, Ž., Agarwal, V., Visentin, D., Ledsam, J.R., Grabska-Barwinska, A., Taylor, K.R., Assael, Y., Jumper, J., Kohli, P., Kelley, D.R.: Effective gene expression prediction from sequence by integrating long-range interactions. Nature Methods **18**(10), 1196–1203 (2021). https://doi.org/10.1038/s41592-021-01252-x
3. Hilmarsson, H., Kumar, A.S., Jia, R., Iyer, R., Bustamante, C.D.: High resolution ancestry deconvolution for next generation genomic data. Nature Communications **13**, 2512 (2022). https://doi.org/10.1038/s41467-022-29936-w
4. Ji, Y., Zhou, Z., Liu, H., Davuluri, R.V.: DNABERT: Pre-trained bidirectional encoder representations from transformers model for DNA-language in genome. In: Bioinformatics. vol. 37, pp. 2112–2120 (2021). https://doi.org/10.1093/bioinformatics/btab083
5. van der Maaten, L., Hinton, G.: Visualizing data using t-SNE. Journal of Machine Learning Research **9**, 2579–2605 (2008)
6. Nguyen, E., Poli, M., Faez, M., Thomas, A.W., Wornow, M., Birber, C., Massaroli, S., Patel, A., Rabideau, C., Bengio, Y., Ermon, S., Ré, C., Massaroli, S.: HyenaDNA: Long-range genomic sequence modeling at single nucleotide resolution. In: Advances in Neural Information Processing Systems. vol. 36. Curran Associates, Inc. (2023)
7. Patterson, N., Price, A.L., Reich, D.: Population structure and eigenanalysis. PLoS Genetics **2**(12), e190 (2006). https://doi.org/10.1371/journal.pgen.0020190
8. Poli, M., Massaroli, S., Nguyen, E., Fu, D.Y., Dao, T., Baccus, S., Bengio, Y., Ermon, S., Ré, C.: Hyena hierarchy: Towards larger convolutional language models. In: Proceedings of the 40th International Conference on Machine Learning. ICML'23, PMLR (2023)
9. Su, J., Lu, Y., Pan, S., Murtadha, A., Wen, B., Liu, Y.: RoFormer: Enhanced transformer with rotary position embedding. In: Neurocomputing. vol. 568, p. 127063 (2024)
10. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: Advances in Neural Information Processing Systems. vol. 30. Curran Associates, Inc. (2017)

Construcción de una teoría didáctica para programación (resumen extendido)

Federico Gómez

Doctorado en Informática - PEDECIBA
Instituto de Computación – Facultad de Ingeniería
fgfrois@fing.edu.uy

1 Introducción

Tomando como punto de partida el trabajo de James Hiebert y James W. Stigler [1], quienes proponen un enfoque novedoso para guiar la construcción de teorías didácticas, basado en un concepto que ellos denominan *Sustained Learning Oportunities* (oportunidades sostenidas de aprendizaje, en adelante SLO), el tema principal del trabajo consiste en investigar los elementos relevantes para crear una teoría didáctica de *programación*. Las SLO constituyen una construcción mediadora entre los procesos de enseñanza y aprendizaje: en lugar de concebir la enseñanza como causa directa del aprendizaje, proponen que el rol docente debe centrarse en crear condiciones adecuadas para *fomentar* el aprendizaje. En torno a las SLO, los autores proponen dos sistemas que denominan *Sistema 1: Enseñanza* y *Sistema 2: Aprendizaje* (en adelante, sistemas S1 y S2), como ilustran en la siguiente figura de [1]:

Fig. 1. Sistemas S1 y S2.

En forma general, definen la primera caja (*Curriculam and teaching resources*) como el conjunto de recursos utilizados por los docentes para sus clases, incluyendo diseño curricular, técnicas de enseñanza, conocimiento didáctico del contenido y la segunda caja (*SLO*) como sistemas temporalmente estables que emergen durante las clases a partir de las interacciones de múltiples variables para crear los contextos en los que se produce el aprendizaje. En cuanto a la flecha (*Teaching*), la definen como el proceso de planificar una lección (sobre algún contenido), implementarla en clase y reflexionar sobre los resultados de dicha implementación. En resumen, el sistema S1 consiste en el proceso de transformación del diseño curricular y otros recursos educativos en SLO específicas enlazadas con objetivos de aprendizaje vinculados al contenido en cuestión.

El trabajo se enfoca específicamente en el sistema S1, con el objetivo de realizar una primera instanciación del mismo a la disciplina *programación*. La investigación busca definir los elementos que permitan diseñar secuencias didácticas capaces de generar SLO específicas para el aprendizaje de conceptos de programación y se nutre de los siguientes elementos:

- La noción de *didáctica* como área específica de un dominio de conocimiento, que busca responder cuatro preguntas fundamentales: qué, cómo, por qué y para quién (educar en dicho dominio).
- Los resultados de las investigaciones epistemológicas (de aquí en adelante, R.I.E) llevadas a cabo por el GIDI (*Grupo de investigación en didáctica de la informática*), acerca de la construcción de conocimiento sobre algoritmos, estructuras de datos y programas.
- La incorporación de ideas fundamentales en informática a la didáctica de las ciencias (en este caso, ciencia de la computación), surgida del proyecto PIMCEU "*El paradigma de las ciencias computacionales y la educación*" conducido por el GIDI durante 2022 y 2023.
- Dos teorías que, por su naturaleza, presentan características compatibles con los R.I.E y brindan aportes importantes para diseñar secuencias didácticas que introduzcan conceptos de programación: la *Teoría de las variaciones* [2] y la *Teoría de las situaciones didácticas* [3].
- Un caso concreto de instanciación del sistema S1, consistente en un estudio empírico que incluye el diseño de una secuencia didáctica (guiado por los elementos previos) para introducir un tema específico de programación, su implementación en clase y un análisis de sus resultados.

2 Hipótesis de la investigación

La hipótesis central adoptada es la denominada *struggle-first*, propuesta por Hiebert y Stigler [1]: los estudiantes crean conexiones entre conceptos de modo efectivo si primero tienen que "luchar" (struggle) de manera productiva con un nuevo concepto, previo a recibir cualquier instrucción formal sobre el mismo, pero contando con conocimiento no formal relacionado. Esta hipótesis se combina con nuestros R.I.E., basados en la Epistemología Genética de Piaget, que describe tres etapas de construcción del conocimiento: *intra*, *inter* y *trans*. Esto posibilita el diseño de actividades que fomenten la lucha productiva en los estudiantes, particularmente en el pasaje de la etapa *intra* a la etapa *inter*. El planteo de actividades iniciales que insten a los estudiantes a interactuar con instancias cotidianas de problemas algorítmicos (etapa *intra*) y actividades posteriores que guíen la conceptualización de los algoritmos que las resuelven (etapa *inter*) estimula la creación de conexiones entre conceptos, previo a su formalización en un lenguaje de programación (etapa *trans*).

3 Desarrollo de marco teórico

La construcción de una teoría didáctica para educar en cualquier dominio debería brindar respuestas a las cuatro preguntas fundamentales. En el caso de la disciplina programación, la revisión bibliográfica realizada en el PIMCEU arrojó elementos para responder a las preguntas *qué* (ideas fundamentales en informática), *para quién* (estudiantes de educación secundaria superior y universitaria inicial) y *por qué* (desarrollar competencias computacionales en los estudiantes para el trabajo científico en el nuevo paradigma). En cuanto a *cómo*, la investigación busca responder esta pregunta a partir de la integración de nuestros R.I.E con la teoría de las variaciones y la teoría de las situaciones didácticas. Se espera obtener, de dicha integración, fundamentos para instanciar la flecha (*Teaching*) del sistema S1 y caracterizar qué SLO resultan adecuadas para el aprendizaje de conceptos de programación.

La teoría de las variaciones [2] postula que el aprendizaje surge de la experimentación de variaciones sucesivas de los *aspectos críticos* (así denominados en la teoría) de los conceptos a construir, en tanto la teoría de las situaciones didácticas [3] propone una metodología de trabajo en el aula que se basa en dos tipos de interacciones: situación *adidáctica* y situación *didáctica* (así denominadas en la teoría). La primera supone la interacción del estudiante con una problemática concreta, a partir de la cual se espera que comience a construir conocimiento acerca de un nuevo concepto, sin intervención del docente. La segunda se da posteriormente, con la intervención del docente, en la cual se explicitan tanto el nuevo concepto introducido en la situación adidáctica como su formulación culturizada (formalización). Se espera que uno de los resultados de la investigación sea una integración entre ambas teorías y nuestros R.I.E. para caracterizar las SLO propias de la enseñanza de programación.

4 Estudio empírico y trabajo restante

Se llevó a cabo un estudio empírico cualitativo consistente en el diseño, implementación en el aula y análisis de una secuencia didáctica para introducir un tema específico de programación. El trabajo se guió por una serie de recomendaciones dadas por Hiebert y Stigler, para el diseño e implementación de secuencias didácticas. Se diseñó una primera versión de la secuencia y se implementó en un grupo correspondiente a un curso universitario inicial de programación imperativa. Se realizó un análisis posterior que posibilitó una serie de ajustes y mejoras a las actividades y luego se implementó en un segundo grupo. Los temas vistos eran los mismos en ambos grupos y ninguno de sus estudiantes había estudiado programación previo a comenzar el curso.

La secuencia introduce una estructura de datos llamada *arreglo con tope* (desconocida aún para los estudiantes) junto con cuatro operaciones básicas para su manipulación: inicialización, inserción, listado y búsqueda. El tope es una variable especial que lleva cuenta de la cantidad de valores almacenados. A modo ilustrativo, se presenta una posible definición de la estructura en lenguaje Pascal (la secuencia prevé que los estudiantes construyan su propia definición):

```
TYPE ArregloConTope = RECORD
                        arreglo : ARRAY[1..N] OF Integer;
                        tope : 0..N;
                      END;
```

Se definieron cuatro grupos de actividades para la secuencia didáctica:

A. Manipulación de una instancia concreta de la estructura de datos.
B. Formalización de la estructura en un lenguaje de programación.
C. Implementación de las operaciones de inicialización e inserción.
D. Implementación de las operaciones de listado y búsqueda.

Las actividades del grupo A parten del conocimiento de los estudiantes en la etapa *intra* y guían, de manera progresiva, su transformación en conocimiento en la etapa *inter* sobre la estructura de datos y las operaciones de inicialización e inserción. Se propuso una situación adidáctica que plantea una estantería en la cual se van colocando discos de izquierda a derecha y se desea saber cuántos hay colocados sin necesidad de contarlos uno por uno, estimulando a los estudiantes a proponer soluciones (*struggle-first*). La situación fue diseñada previendo que, tras evaluar alternativas, propongan una solución que involucre usar algún "papel" o "cartel" para llevar cuenta de los discos, el cual será formalizado como tope en las actividades del grupo B, resultando un elemento esencial (aspecto crítico) para una adecuada comprensión de la estructura.

Las actividades del grupo B inducen establecer una correspondencia entre la instancia concreta de la estructura de datos (la estantería) y su contraparte en el lenguaje de programación (lenguaje formal, etapa *trans*). Estimulan a utilizar conocimientos previos y adaptarlos para proponer una definición en un lenguaje de programación para la nueva estructura (arreglo con tope), analizando sus similitudes y diferencias con la estructura de arreglo clásico (ya conocida por los estudiantes), para estimular a tomar conciencia del tope como aspecto crítico.

Finalmente, las actividades de los grupos C y D piden escribir, compilar y ejecutar subprogramas que implementen las cuatro operaciones solicitadas, en base al conocimiento construido en las actividades previas, lo que profundiza su transformación hacia conocimiento en la etapa *trans*.

Tras completar el estudio, un primer análisis cualitativo arrojó que los ajustes y mejoras al diseño de las actividades, tras su implementación en el primer grupo, tuvieron un impacto positivo en la generación de situaciones de aprendizaje (las SLO del sistema S1) durante la implementación en el segundo grupo. La evidencia recolectada en ambos (dada por las hojas de papel y archivos fuente con las respuestas de los estudiantes) y las observaciones realizadas en ambas instancias sugieren que los estudiantes del segundo grupo construyeron conceptos con mayor solidez, tanto sobre la estructura de datos como sobre las operaciones. De todos modos, en ambos grupos se constató que el carácter gradual de las actividades (pensadas para guiar la construcción progresiva de los conceptos) permitió a los estudiantes completarlas con éxito. La tarea principal del docente fue conducir el desarrollo de las actividades (en lugar de tener un rol expositivo), guiar discusiones colectivas y realizar puntualizaciones cuando fue necesario. A la fecha, se continúa trabajando en el análisis en profundidad de los resultados del estudio empírico y obtención de conclusiones del mismo, con miras a sintetizar los fundamentos teóricos que proveen la base inicial para la construcción de la teoría didáctica.

Referencias

1. Hiebert, J. & Stigler, J.W., Creating Practical Theories of Teaching. En: Praetorius, A.K., Charalambous, C.Y., Theorizing Teaching: Current Status and Open Issues. Springer (2023).
2. Marton, F. Necessary Conditions of Learning. Routledge (2015).
3. Brousseau, G., Iniciación al estudio de la Teoría de las Situaciones Didácticas (trad. Español). Buenos Aires: Libros del Zorzal (2007).

Iniciativas de I+D en SeCIU/RAU

Santiago Elizondo[1,2], Leonardo Alberro[3], Mario Guerri[1,2], Martín Giachino[3], Leonardo Vidal[3], and Eduardo Grampín[1,3]

[1] Red Académica Uruguaya, Uruguay
[2] Servicio Central de Informática Universitaria, Udelar, Uruguay
{santiago.elizondo, mario.guerri}@seciu.edu.uy
[3] Instituto de Computación, Facultad de Ingeniería, Udelar, Uruguay
{lalberro, giachino, lvidal, grampin}@fing.edu.uy

Abstract. Se presentan algunos proyectos llevados a cabo en el marco del convenio de colaboración entre Fing y SeCIU/RAU, que buscan fortalecer la infraestructura tecnológica y los servicios de conectividad de la red académica nacional, contribuyendo a la mejora de los servicios académicos y de investigación de la Udelar.

Keywords: Acceso Inalámbrico de Calidad · Datacenter avanzado · Evolución de la Red Académica.

1 Introducción

En el marco de un convenio de colaboración entre la Red Académica Uruguaya (RAU)[10], gestionada por el Servicio Central de Informática Universitaria (SeCIU)[12], y el Instituto de Computación (INCO)[7] de la Facultad de Ingeniería (Fing)[3] de la Universidad de la República (Udelar)[13], se impulsaron diversas iniciativas de investigación y desarrollo orientadas a fortalecer la infraestructura tecnológica y los servicios de conectividad de la red académica nacional.

El propósito de este convenio es promover la cooperación entre los equipos técnicos de la RAU y los grupos de investigación de la Facultad, generando sinergias que permitan abordar desafíos técnicos relevantes para la comunidad universitaria.

En este trabajo se presentan tres de los proyectos desarrollados en este contexto:

- **Rediseño de la red del datacenter de SeCIU**, orientado a modernizar la arquitectura de red y mejorar su escalabilidad.
- **AIRE (Acceso Inalámbrico a las REdes académicas)**, una iniciativa para ofrecer conectividad Wi-Fi de calidad y gestionada centralmente en los distintos servicios universitarios.
- **Diseño del nuevo backbone de la RAU**, enfocado en la evolución de la red troncal hacia mayores capacidades y una cobertura nacional más amplia.

Cada uno de estos proyectos aborda aspectos distintos pero complementarios de la evolución tecnológica de la RAU, contribuyendo a la mejora continua de los servicios académicos y de investigación de la Udelar.

2 Rediseño de la red del datacenter

Desde hace algunos años, el equipo técnico de la RAU junto con investigadores del grupo MINA[6] comenzaron a llevar adelante un proyecto con el fin de diseñar una nueva arquitectura de red que modernice y aumente las capacidades del datacenter de SeCIU. El objetivo era tener una solución que resolviera las necesidades actuales y que contara con buena escalabilidad hacia el futuro.

En este marco, se tuvo en cuenta la tendencia actual de implementar datacenters siguiendo topologías que favorecen la agregación del ancho de banda entre servidores, resolviendo eficientemente las necesidades del tráfico este-oeste. Bajo estas premisas se puso foco en las topologías conocidas como "Fat-Tree"[15], las cuales brindan múltiples caminos entre servidores, utilizando nodos con capacidad de capa 3 que implementen protocolos de enrutamiento con soporte de multi-caminos (ECMP[17], por sus siglas en inglés "Equal Cost Multi Path"). Con este diseño se buscaba utilizar equipos sencillos y con iguales características en cada una de sus capas, evitando la necesidad de adquirir switches de agregación de gran porte. A nivel de capa 2, era deseable el soporte de una red Ethernet superpuesta (overlay), particularmente implementada con la tecnología VXLAN[18] como plano de datos y EVPN[16] de MP-BGP como plano de control.

La escala del datacenter y la realidad del mercado con ausencia de switches de pequeño porte que tuvieran capacidad de implementar los protocolos requeridos, llevó a ajustar el diseño original para pasar a una arquitectura spine-leaf.

La aplicación de las tecnologías VXLAN/EVPN en el nuevo diseño permite pasar el enrutamiento centralizado a un formato distribuido, sacando mayor provecho de los múltiples caminos. Este cambio desencadenó la necesidad de ajustar la forma en que se aplican los controles de seguridad en las conexiones entre servidores. Hasta el momento estos controles también siguen un esquema centralizado, por lo que es necesario realizar un nuevo análisis que permita determinar la mejor solución para distribuir la seguridad del tráfico este-oeste.

3 AIRE: Acceso Inalámbrico a las REdes académicas

Dentro de la Udelar comenzaron a detectarse casos de Access Points trabajando de forma autónoma y publicando redes Wi-Fi sin un criterio definido ni una gestión centralizada. Estas implementaciones generan soluciones Wi-Fi fragmentadas, donde cada punto de acceso ofrece una conectividad aislada, sin posibilitar la movilidad del usuario dentro del centro universitario y con una gestión compleja cuando surgen problemas de cobertura, interferencias u otros incidentes. Al mismo tiempo, el aumento de dispositivos de los usuarios que requieren conexión inalámbrica provoca una alta demanda de conectividad Wi-Fi dentro de la Udelar, generando que SeCIU reciba diversos pedidos de ayuda para diseñar despliegues de soluciones Wi-Fi de calidad. En respuesta a esta situación, se conformó un grupo de trabajo con investigadores de la Fing y funcionarios técnicos de SeCIU con el fin de definir lineamientos a seguir para poder unificar los criterios al implementar soluciones Wi-Fi en la Udelar.

3.1 Objetivo que se persigue

Brindar acceso inalámbrico de calidad a toda la comunidad universitaria en los servicios universitarios con las siguientes características: seguridad, control de acceso, calidad adecuada para los requerimientos de educación, investigación y gestión, todo esto con una amplia cobertura *indoor* y *outdoor*, en una infraestructura de red gestionada centralmente. Este servicio será utilizado por docentes, investigadores, funcionarios, estudiantes o visitantes que se encuentren en el lugar. Además, las personas que trabajan en un servicio, deben tener la posibilidad de hacer uso de la red con sus mismas credenciales cuando están en otro servicio, y los profesores visitantes extranjeros podrán acceder a la red utilizando las credenciales de su institución de origen.

4 Diseño del nuevo backbone de la RAU

El proyecto parte de la realidad del backbone actual de la RAU, proponiéndose en el corto plazo ampliar las capacidades de los enlaces sin alterar la topología actual. En el mediano plazo se propone comenzar a desarrollar puntos de presencia en todo el territorio, teniendo en cuenta la distribución de los Centros Universitarios Regionales (CENUR) de la Udelar y otras instituciones (por ejemplo la Universidad Tecnológica del Uruguay (UTEC)[14] y la Dirección General de Educación Técnico Profesional – UTU (DGETP-UTU)[2]).

4.1 Corto plazo

La principal motivación es que se están alcanzando niveles de tráfico cercanos a la saturación de los enlaces troncales y de Internet de 1Gbps, además de que la infraestructura actual está limitada a 300Mbps en los sitios remotos (servicios).

El objetivo es llegar a 10Gbps en los enlaces troncales y de acceso a Internet, y hasta 1Gbps en los servicios.

4.2 Mediano/largo plazo

Evolución a un backbone nacional con más de 10 puntos de presencia en el área metropolitana e interior del país con conectividad de hasta 100Gbps. Se diseñó una topología tentativa en base a conversaciones con la Administración Nacional de Telecomunicaciones (ANTEL)[1] y la Cooperación Latino Americana de Redes Avanzadas (RedCLARA)[11], que se irá revisando teniendo en cuenta la presencia de sitios de la UTEC y otras instituciones en el interior y el área metropolitana.

El objetivo es desplegar esta topología en entornos emulados/simulados y experimentar con el plano de control (enrutamiento) y el plano de datos (tráfico), cosiderando principalmente los enlaces troncales de 100Gbps, y hasta 10Gbps en los servicios.

Las herramientas utilizadas son el emulador GNS3[5] para la validación funcional del plano de control, es decir, toda la configuración de la red, incluyendo fundamentalmente el enrutamiento, con la base de código de FRR[4], y simuladores de eventos discretos para las pruebas de tráfico; estamos evaluado el uso de ns-3[8] y/u OMNeT++[9].

5 Conclusión y trabajo futuro

La cooperación de Fing con SeCIU/RAU en proyectos de I+D ha permitido resolver problemas concretos de la infraestructura que da soporte a los servicios de la comunidad académica. El vínculo establecido se ha mantenido durante los últimos 5 años, y avizoramos mantener la cooperación en el futuro.

References

1. Administración Nacional de Telecomunicaciones (ANTEL). https://antel.com.uy, Último acceso: 03 Dic 2025
2. Dirección General de Educación Técnico Profesional – UTU (DGETP-UTU). https://www.utu.edu.uy/, Último acceso: 03 Dic 2025
3. Facultad de Ingeniería (Fing), Universidad de la República. https://www.fing.edu.uy, Último acceso: 03 Dic 2025
4. FRRouting Project (FRR). https://frrouting.org/, Último acceso: 03 Dic 2025
5. GNS3 – Graphical Network Simulator-3. https://gns3.com/, Último acceso: 03 Dic 2025
6. Grupo MINA – Network Management / Artificial Intelligence. https://www.fing.edu.uy/inco/grupos/mina/, Último acceso: 03 Dic 2025
7. Instituto de Computación (INCO), Facultad de Ingeniería, Universidad de la República. https://www.fing.edu.uy/inco, Último acceso: 03 Dic 2025
8. ns-3 Network Simulator. https://www.nsnam.org, Último acceso: 03 Dic 2025
9. OMNeT++ Discrete Event Simulator. https://omnetpp.org/, Último acceso: 03 Dic 2025
10. Red Académica Uruguaya (RAU). https://rau.edu.uy, Último acceso: 03 Dic 2025
11. RedCLARA – Cooperación Latino Americana de Redes Avanzadas. https://redclara.net/, Último acceso: 03 Dic 2025
12. Servicio Central de Informática Universitaria (SeCIU). https://www.seciu.edu.uy, Último acceso: 03 Dic 2025
13. Universidad de la República (Udelar). https://udelar.edu.uy, Último acceso: 03 Dic 2025
14. Universidad Tecnológica del Uruguay (UTEC). https://utec.edu.uy/, Último acceso: 03 Dic 2025
15. Al-Fares, M., Loukissas, A., Vahdat, A.: A scalable, commodity data center network architecture. p. 63–74. SIGCOMM '08, Association for Computing Machinery, New York, NY, USA (2008). https://doi.org/10.1145/1402958.1402967
16. Drake, J., Henderickx, W., Sajassi, A., Aggarwal, R., Bitar, D.N.N., Isaac, A., Uttaro, J.: BGP MPLS-Based Ethernet VPN. RFC 7432 (Feb 2015). https://doi.org/10.17487/RFC7432, https://www.rfc-editor.org/info/rfc7432
17. Hopps, C.: Analysis of an Equal-Cost Multi-Path Algorithm. RFC 2992 (Nov 2000). https://doi.org/10.17487/RFC2992, https://www.rfc-editor.org/info/rfc2992
18. Mahalingam, M., Dutt, D., Duda, K., Agarwal, P., Kreeger, L., Sridhar, T., Bursell, M., Wright, C.: Virtual eXtensible Local Area Network (VXLAN): A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks. RFC 7348 (Aug 2014). https://doi.org/10.17487/RFC7348, https://www.rfc-editor.org/info/rfc7348

Redes neuronales para modelado y predicción de polución en entornos urbanos

Josefina Cardozo[0009−0000−8263−0994] and
Sergio Nesmachnow[0000−0002−8146−4012]

Universidad de la República, Uruguay
{josefina.cardozo,sergion}@fing.edu.uy

Resumen La contaminación del aire es uno de los principales factores de riesgo ambiental para la salud humana. En Europa, las partículas finas ($PM_{2,5}$), el dióxido de nitrógeno (NO_2) y el ozono (O_3) son los contaminantes responsables de la mayoría de las muertes prematuras asociadas a la contaminación atmosférica. Por lo tanto, la predicción a corto plazo de las concentraciones de estos contaminantes es de gran importancia para la salud humana. En este trabajo se propone un modelo de red neuronal recurrente tipo Long Short-Term Memory (LSTM) con arquitectura codificador–decodificador para pronosticar concentraciones de PM2,5, NO_2 y O_3 a 1 y 24 horas en la ciudad de Madrid, utilizando observaciones de 24 estaciones de monitoreo. Los resultados se comparan con un modelo base de perceptrón multicapa (MLP), mostrando que el LSTM reduce el error absoluto medio (MAE) hasta en un 60 % a una hora y entre un 11–17 % a 24 horas, con reducciones similares en la Raíz del Error Cuadrático Medio (RMSE).

Keywords: aprendizaje profundo · LSTM · codificador-decodificador · MLP · aprendizaje supervisado · modelo de predicción

1. Introducción

Los modelos tradicionales lineales o estadísticos proporcionan referencias útiles, entre ellos el modelo de regresión logística [1], ARIMA [3], SARIMA [2], entre otros. No obstante, estos métodos presentan dificultades debido a la complejidad de las dependencias no lineales y espaciotemporales en los datos de series temporales. En esta línea de trabajo, este artículo presenta un enfoque basado en redes neuronales multicapa, específicamente una arquitectura recurrente tipo codificador-decodificador LSTM, para predecir las concentraciones de contaminantes atmosféricos. Se aborda un caso de estudio real: el modelado y la predicción de la contaminación del aire en Madrid, España, utilizando datos abiertos reales [5].

Los datos multivariados de series temporales permiten modelar tanto la dinámica temporal como las interacciones entre los contaminantes y las condiciones meteorológicas. Los estudios existentes suelen centrarse en una única estación o en un solo contaminante, o no consideran los factores externos que influyen, lo que limita su capacidad predictiva. Los principales resultados y contribuciones de esta investigación incluyen el uso de observaciones horarias provenientes

de la red abierta de calidad del aire de Madrid, España —que abarca $PM_{2,5}$, NO_2 y O_3, junto con variables meteorológicas— para desarrollar y evaluar un modelo codificador-decodificador LSTM capaz de predecir simultáneamente las concentraciones de estos contaminantes con un horizonte de una hora y de 24 horas.

2. Predicción multivariante de contaminación a corto plazo

2.1. Descripción del problema

El estudio aborda la predicción supervisada multivariante de series temporales de la calidad del aire urbano, utilizando datos y variables meteorológicas. El problema se define de la siguiente manera.

Sea $x_t \in \mathbb{R}^F$ el vector de características horarias en el instante t, y sea $y_t \in \mathbb{R}^P$ el vector objetivo de concentraciones de los contaminantes ($P = 3$: $PM_{2,5}$, NO_2, O_3). Usando las pasadas W horas como datos de entrada, se define $X_{t-W+1:t} = (x_{t-W+1}, \ldots, x_t)$. El objetivo del problema es aprender una función f_θ que prediga concentraciones futuras a horizontes $\tau \in \{1, 24\}$ horas: $\hat{y}_{t+\tau} = f_\theta\big(X_{t-W+1:t}\big), \tau \in \{1, 24\}$. Las salidas correspondientes a los contaminantes objetivo se consideran de forma simultánea, y los modelos se entrenan y evalúan individualmente en cada estación de monitoreo.

2.2. Caso de estudio

El caso de estudio corresponde a la ciudad de Madrid, España. La red de monitoreo tiene 37 estaciones distribuidas en entornos de tráfico urbano, fondo urbano y suburbanos.

Se analizan observaciones horarias comprendidas entre el 1 de julio de 2021 y el 31 de diciembre de 2024, excluyendo los períodos de confinamiento por COVID-19 (5 de marzo–21 de junio de 2020; 25 de octubre de 2020–9 de mayo de 2021) para evitar episodios atípicos. En este estudio, los conjuntos de datos se extraen del Portal de Datos Abiertos de Madrid, donde se encuentran disponibles mediciones horarias. Las variables consideradas incluyen contaminantes atmosféricos —PM10, PM2,5, CO, NO_2, SO_2, y O_3—y covariables meteorológicas: velocidad y dirección del viento, temperatura del aire, presión atmosférica, radiación solar y precipitación.

3. Metodología

El enfoque metodológico propuesto se detalla a continuación. El conjunto de datos utiliza registros horarios de concentraciones de contaminantes atmosféricos y variables meteorológicas de la red abierta de calidad del aire de Madrid [5]. Las series fueron integradas por código de estación y marca temporal. Se eliminaron

campos administrativos irrelevantes, se depuraron valores atípicos según umbrales regulatorios y los valores faltantes se trataron mediante imputación acompañada de marcadores implícitos, preservando la continuidad temporal. Todas las variables fueron normalizadas utilizando los parámetros del conjunto de entrenamiento, evitando así filtraciones de información. Los datos se estructuraron en ventanas deslizantes para generar las secuencias de entrada y salida requeridas por los modelos.

Para la comparación de resultados se consideró un modelo base no recurrente, un MLP con una arquitectura simple, que recibe cada ventana de entrada $X \in \mathbb{R}^{W \times F}$, la aplana y la procesa mediante capas densas para generar las predicciones de salida. Este modelo es una referencia sin capacidad explícita de memoria temporal.

El modelo principal tiene una arquitectura encoder–decoder basada en redes neuronales recurrentes del tipo LSTM. El codificador procesa la secuencia de entrada y resume la información en un vector de contexto junto con sus estados internos (h, c). Este contexto se replica mediante una capa RepeatVector para generar una secuencia de longitud igual al horizonte de predicción. El decodificador, también un LSTM inicializado con los estados finales del codificador, genera las predicciones paso a paso, que son transformadas mediante una capa TimeDistributed(Dense) que genera las concentraciones estimadas $\hat{Y} \in \mathbb{R}^{H \times P}$ de los tres contaminantes. Ambas capas recurrentes utilizan regularización $\ell_2(10^{-4})$ para evitar el sobreajuste.

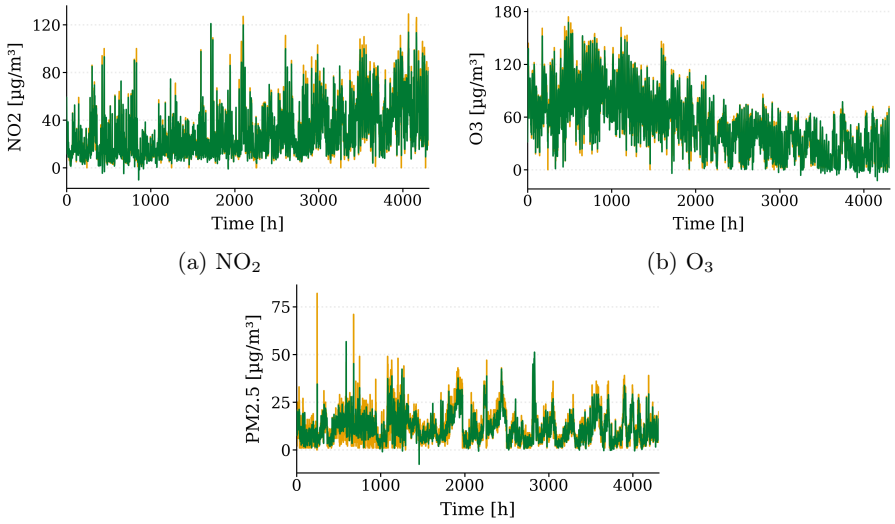
4. Validación experimental

Para el entrenamiento y evaluación de los modelos, se implementó un esquema de ventana deslizante mediante la clase WindowGenerator, que generó automáticamente las secuencias de entrada y salida, así como los conjuntos de entrenamiento, validación y prueba. Los modelos fueron entrenados con el optimizador Adam (tasa de aprendizaje 10^{-5}) durante un máximo de 200 épocas, aplicando early stopping sobre la pérdida de validación (paciencia = 5). La función de pérdida fue el MAE, complementada con el error RMSE como métrica adicional.

Se aplicó validación cruzada temporal con tres particiones consecutivas, cada una con 18 meses de entrenamiento, 6 meses de validación y un período final de prueba. Este enfoque siguió un orden cronológico, evitando que se filtraran comportamientos futuros y permitiendo la evaluación en distintos intervalos temporales. De esta manera, se evaluó la estabilidad del desempeño del modelo frente a diferentes períodos de la serie. Todos los experimentos se realizaron en la infraestructura del Centro Nacional de Supercomputación (Cluster-UY), Uruguay [4].

La Figura 1 muestra las predicciones a una hora del modelo LSTM. Corresponde a la estación urbana de tráfico Escuelas Aguirre, seleccionada porque mide simultáneamente los tres contaminantes y las variables meteorológicas, generando una serie multivariante completa. Entre los contaminantes, $PM_{2,5}$ presentó

mayores desviaciones respecto a los valores observados, probablemente debido a su alta variabilidad. En cambio, NO_2 y O_3 reprodujeron con mayor exactitud las tendencias y magnitudes observadas. La arquitectura utilizada prioriza las predicciones horarias estables y la tendencia general del comportamiento del contaminante, por lo que los episodios abruptos, debido a su menor frecuencia, no son el foco del modelo.



(a) NO_2 (b) O_3

(c) $PM_{2,5}$

Figura 1: Predicciones a una hora del modelo LSTM *encoder–decoder* en la estación Escuelas Aguirre. (—: observadas, —: predichas).

Las Tablas 1 y 2 resumen los resultados para horizontes de 1 y 24 horas. Los errores aumentaron con el horizonte de predicción: MAE y RMSE fueron consistentemente mayores a 24 horas que a 1 hora. En promedio, el modelo LSTM redujo el MAE entre un 39 % y 60 % y el RMSE entre un 33 % y 56 % respecto al MLP para el horizonte de una hora. Para 24 horas, las reducciones están entre 11–17 % en MAE y 9–15 % en RMSE.

La magnitud de los errores depende de cada contaminante y se interpreta en relación con sus valores típicos. En el caso de NO_2, la concentración media es de 26.92 $\mu g/m^3$ y puede variar por el tráfico urbano. El valor del MAE de aproximadamente 6 $\mu g/m^3$ representa un error relativo moderado. El contaminante O_3 tiene una media de 54.37 $\mu g/m^3$ y los errores obtenidos muestran que el modelo logra seguir bien su tendencia general. El caso de $PM_{2.5}$ es distinto porque su media es más baja (9.77 $\mu g/m^3$), pero tiene una variabilidad muy alta. Esta inestabilidad hace que sea un contaminante más difícil de predecir. Los valores

obtenidos (MAE de 3.27 μg/m^3 a 1 h y 5.42 μg/m^3 a 24 h) son coherentes con su nivel de variabilidad.

Tabla 1: Resultados experimentales – horizonte 1 h en la estación 8 (μg/m^3).

	$PM_{2.5}$		NO_2		O_3	
Modelo	MAE	RMSE	MAE	RMSE	MAE	RMSE
LSTM	3.27±0.30	4.43±0.51	6.09±0.17	8.70±0.19	6.39±0.49	8.73±0.68
MLP	9.02±0.29	11.22±0.26	10.04±2.92	12.97±3.32	16.34±7.26	19.78±8.20

Tabla 2: Resultados experimentales – horizonte 24 h en la estación 8 (μg/m^3).

	$PM_{2.5}$		NO_2		O_3	
Modelo	MAE	RMSE	MAE	RMSE	MAE	RMSE
LSTM	5.42±0.23	6.95±0.36	13.72±1.05	17.86±1.29	15.97±1.46	19.62±1.82
MLP	6.52±0.56	8.19±0.70	15.54±1.47	19.58±1.73	18.44±2.03	22.46±2.50

En términos de tiempo de entrenamiento, el LSTM necesitó, en promedio, 120 épocas para converger en el horizonte de 1 h y 100 épocas de 45 segundos para 24 horas. El MLP, en cambio, requirió solo 3 minutos, pero con resultados significativamente inferiores. El modelo propuesto logró mejorar de forma consistente la precisión de las predicciones sin comprometer la estabilidad del entrenamiento.

Referencias

1. Aditya, C., Deshmukh, C., Nayana, D., Gandhi, P.: Detection and prediction of air pollution using machine learning models. International Journal of Engineering Trends and Technology **59**(4), 204–207 (May 2018). https://doi.org/10.14445/22315381/IJETT-V59P238
2. Deepan, S., Saravanan, M.: Air quality index prediction using seasonal autoregressive integrated moving average transductive long short-term memory. ETRI Journal **46**(5), 915–927 (2024). https://doi.org/10.4218/etrij.2023-0283
3. Marinov, E., Petrova-Antonova, D., Malinov, S.: Time series forecasting of air quality: A case study of Sofia city. Atmosphere **13**(5), 788 (2022). https://doi.org/10.3390/atmos13050788
4. Nesmachnow, S., Iturriaga, S.: Cluster-UY: Collaborative scientific high performance computing in Uruguay. In: Torres, M., Klapp, J. (eds.) Supercomputing. ISUM 2019. Communications in Computer and Information Science, vol. 1151. Springer, Cham (2019)

5. Toutouh, J., Lebrusán, I., Nesmachnow, S.: Computational Intelligence for Evaluating the Air Quality in the Center of Madrid, Spain, pp. 115–127. Springer International Publishing (2020). https://doi.org/10.1007/978-3-030-41913-4_10

Crystal, the unbureaucratic programming language
Extended Abstract

Beta Ziliani[1] [0000−0001−7071−6010]

ORT University, Montevideo, Uruguay ziliani@ort.edu.uy

In this paper, we position Crystal[2] within the landscape of programming languages, and argue that it's an interesting playground for interesting research projects.

In two lines, Crystal is *statically typed* and *compiled*, like Java and C++, but free of bureaucracy, similar like Ruby or Python. In order to achieve that:

Crystal trades modularity for expressivity

Is this very essence that makes Crystal a unique interesting subject of study.

1 Crystal in a nutshell

Crystal is an open-source programming language borned in Argentina that's been around for more than a decade, with several companies worldwide relying on it for varying applications.[3] It has a syntax inspired by Ruby, and much of Crystal code resembles Ruby. However, it has a *static* type system, like C++ and Java, where non-sensical programs are rejected by the typechecker, but without their verbosity. Like these languages, Crystal is compiled to efficient native code, taking advantage of the types and using the LLVM compiler infrastructure. As a result, it produces programs that are at the same time declarative, fast, and memory-efficient.[1]

2 Unbureaucratic

The main attraction of the language is that most types are inferred by the typechecker. This enables rapid prototyping without sacrificing the safety provided by types. Notably, Crystal offers two capabilities that are rarely found in a static language: *duck typing* and *monkey patching*. In this abstract we focus on the first feature, as it sufficiently illustrates Crystal's gains and pains.

2.1 Duck typing

The following code is valid in both Ruby and Crystal:

```
1  def adder(x, y)
2    x + y
3  end
4
5  adder 1, 2 # => 3
6  adder "hello", " world" # => hello world
```

It demonstrates duck typing, a concept coined after the phrase:

If it walks like a duck and it quacks like a duck, then it must be a duck

Applying this analogy, our code is valid because, *if it can be added, then it's added*. Since both numbers and strings support the + operator, they can be added. The crucial difference from Ruby, however, is that if we try to add two incompatible objects, Crystal catches the error at compile time. For that reason, we say that Crystal is a *safe language*:

```
adder 1, " world"
# Error: expected argument #1 to 'Int32#+' to be Float32, ..., not String
```

Duck typing happens on an everyday basis in dynamic languages. But in static —or safe— languages like Java or C++, there is no way around types: we must specify that `x` and `y` can be added, either via an interface, or a common base class. In Crystal you *can* use similar concepts, but you are not forced to.

2.2 The bureaucracy is necessary for *most* static languages

Languages like Java and C++ require a ton of bureaucracy —that is, to be explicit about types— for a very good reason: traditional compilers are *modular*, splitting up programs into *units of compilation* (for instance, each class); performing the compilation on each unit in isolation.

But in order to achieve modular compilation, the compiler needs to be sure that local changes in one unit won't impact other units. By having interfaces and declared types in each function, the compiler makes sure that, as long as there is no change in the *public interface* of the unit, then other units depending on it won't have to be recompiled.

3 The power of Crystal, it's curse

As we mentioned at the beginning, Crystal trades modularity for expressivity. That is, every time the compiler runs, it analyses the entire program, including the code of any library in which it depends on. Then, it takes advantage of having this global view of the program to understand the flow of types. On the plus side, as we've seen above, this gives us a lot of freedom to write code in an unbureaucratic way. On the downside, however, this implies that every change in the code, as minimal as it may be, forces a complete re-analysis and compilation of the program.

In practice, the situation is not nearly as bad as it seems. First, Crystal has a very fast compiler (as a matter of fact, it's written in Crystal itself!). So even for medium-sized programs the compilation time is acceptable. Second, Crystal has a *caching mechanism* that re-uses the object code of units that haven't changed from one compilation to the next one. That is, it parses and analyses the code and its types, and produces intermediate LLVM language code. If a unit is byte-to-byte

equal to the cached one, then it's not processed further; it's binary object is re-used. This makes the compilation incremental to some extent, although not as much as we'd like to.

4 The curse of Crystal, our gain

In a preliminary study, we measured that most commits in several Crystal projects reuse more than 90% of the object files from the previous compilation. This opens a venue of research: can we also cache the results of the analysis phase, so that we don't have to re-analyze the entire program from scratch at every compilation?

To understand the difficulty at hand, let's go back to the `adder` example. Let's assume that the `adder` function is in the `adder` module, and the code calling it is in the `main` module. In the compiled code we observe that Crystal specializes the function, creating two versions, one taking integers and another taking strings. These specializations lie in the `adder` module. Then, when we add a third call using a different type (say, two floats), then the compiler needs to generate a third one, and append it to that module. That is, a change in the `main` module forces a change in the `adder` module, breaking the modularity.

When compared to existing typecheckers for dynamic languages, like TypeScript for JavaScript or Pyright for Python, these languages's runtimes are prepared to work with untyped objects, allowing themselves to throw the towel and just tag an object with the `Any` type.[4] Instead, Crystal requires each function call to have a precise (list) of functions to call.

Our first purpose is to study ways in which we can avoid re-anaylising a module for already visited types. For instance, we want to avoid visiting the `adder` module if we repeat a call to the `adder` function with two integers in the `main` module. Following our preliminary study, we expect that only in a few cases we'll have to re-analyze the entire code.

On the longer term, we have several other possibilities of research, starting from a language that decided to trade modularity for expressivity.

References

1. hanabi1224, contributors: Crystal Benchmarks. https://programming-language-benchmarks.vercel.app/crystal, [Accessed 10-11-2025]
2. The Crystal Core Team: Crystal Main Page. https://crystal-lang.org, [Accessed 10-11-2025]
3. The Crystal Core Team: Crystal: Used in Production. https://crystal-lang.org/used_in_prod, [Accessed 10-11-2025]
4. The Pyright Team: Understanding type inference. https://microsoft.github.io/pyright/#/type-inference, [Accessed 10-11-2025]

Asignación de estudiantes a cursos con cupos, considerando preferencias estudiantiles, superposición de horarios y prioridades institucionales

Cristina González[1], Álvaro Martínez[1], Pedro Piñeyro[2], Libertad Tansini[2], and Carlos E. Testuri[2]

[1] SeCIU, Udelar, Uruguay
{cristina.gonzalez,alvaro.martinez}@seciu.edu.uy
[2] InCo, FIng, Udelar, Uruguay
{ppineyro,libertad,ctesturi}@fing.edu.uy

Resumen En este trabajo se describe la solución informática desarrollada para el problema de asignación de estudiantes a cursos con cupos y superposición de horarios en servicios de la Universidad de la República, maximizando la satisfacción de las preferencias estudiantiles indicadas al momento de la inscripción, y considerando la priorización de los diferentes servicios. Para ello fue necesario realizar cambios en los sistemas informáticos de autogestión estudiantil y de bedelías, así como el desarrollo de modelos matemáticos de optimización para maximizar las preferencias estudiantiles bajo las restricciones impuestas por los servicios. Se presentan los resultados obtenidos en la primera experiencia de la solución propuesta para la Facultad de Derecho y la Facultad de Psicología de la Universidad de la República. De estos resultados se destaca la alta satisfacción de las preferencias estudiantiles a las inscripciones realizadas y una mayor disponibilidad de información para el apoyo a la toma de decisiones de las bedelías.

Keywords: Asignación de estudiantes · Matching con preferencias · Gestión universitaria · Programación matemática · Optimización.

1. Introducción

En algunas facultades de la Universidad de la República (Udelar) existen cursos con cupos que no cubren la demanda estudiantil. La inscripción estudiantil a los horarios de los cursos, se realiza a través de una plataforma en Internet, según el orden en que el servidor los atiende y hasta completar el cupo, lo cual tiene varias consecuencias negativas. Los horarios más solicitados se completan rápidamente por quienes acceden primero al servidor. Los estudiantes con mejor acceso a Internet compiten con ventaja al momento de la apertura de las inscripciones. Durante el proceso de inscripción se dan picos de acceso al sistema por la alta demanda, lo que afecta el funcionamiento de la plataforma. Además, no se registra información sobre la demanda insatisfecha.

Para atender la situación descrita anteriormente, el Consejo Directivo Central (CDC) de la Udelar resolvió que para las inscripciones a cursos con cupo no se podrá depender en ningún caso del orden de llegada del estudiante que aspira a inscribirse, sino que deben realizarse con herramientas y procedimientos de distribución definidos adecuadamente [1]. Para cumplir con dicha resolución, el Servicio Central de Informática de la Udelar (SeCIU) trabajó con el asesoramiento del Departamento de Investigación Operativa del Instituto de Computación (InCo) de la Facultad de Ingeniería de la Udelar, en el marco del convenio de cooperación existente entre ambas instituciones. Se inició el trabajo con las facultades que presentan situaciones más críticas en relación a las inscripciones, como son la Facultad de Derecho (FDER) y la Facultad de Psicología (PSICO), las cuales tienen características particulares que requieren criterios distintos para la asignación de cupos. FDER requiere priorizar estudiantes con domicilios distantes al centro de estudio, en los cursos con horarios definidos con modalidad 'A Distancia' e 'Híbrida'. Además, se priorizan estudiantes que trabajen o tengan menores a su cargo. Por su parte, PSICO prioriza el avance en créditos en la carrera y, en caso de empate, otorga prioridad a estudiantes que cuentan con beca. Además, PSICO agrega restricciones de cursado de las unidades curriculares (UCs): un máximo de cuatro optativas, una práctica y un proyecto.

2. Metodología

Como solución, se plantea que, en lugar de que los estudiantes elijan directamente cursos y horarios a los que quieren inscribirse, se les permita establecer preferencias sobre los cursos y sus horarios. A partir de estas preferencias se busca una asignación de la inscripción que maximiza las preferencias totales de los estudiantes considerando los criterios de prioridad, mientras se cumple con los cupos de cursos y horarios, y se evita la superposición de los horarios asignados. Los estudiantes que no son asignados quedan en una lista de suplentes ordenada según los criterios de prioridad.

Los problemas de asignación de estudiantes a actividades en base a sus preferencias se enmarcan dentro del área de problemas de asignación [2] y han sido estudiados hace varias décadas [3], [4]. Ejemplos más recientes consideran restricciones de capacidad y de calidad de la asignación [5], y tienen en cuenta las preferencias tanto de estudiantes como de profesores [6].

Para implementar la solución se desarrollaron modelos de optimización discreta con las particularidades de cada facultad y se adaptó el Sistema de Gestión Administrativa de Enseñanza [7], en el cual los estudiantes gestionan sus inscripciones. Por su parte, las bedelías de las facultades aportan, al sistema, información de los cursos disponibles, con detalle de días y horarios de dictado. Además, para el caso de PSICO se definen grupos de UCs con restricciones de cantidad de cursada. El sistema recaba para cada estudiante las constancias presentadas, el conjunto de cursos en los que se quiere inscribir, sus preferencias de horarios para cada uno, el orden de preferencia entre los cursos y el avance en créditos (estos dos últimos solo para PSICO).

Con toda la información antes mencionada, se resuelve el modelo según facultad y se obtiene para cada estudiante la asignación a uno de los cupos o a la lista de suplentes de un horario de los cursos a los que se inscribió. Dicha asignación establece una distribución equitativa de las preferencias estudiantiles, teniendo en cuenta los criterios de priorización impuestos por cada servicio. Con el resultado proporcionado por el modelo, el sistema realiza la asignación de los cupos y genera las listas de suplentes para cada curso.

3. Resultados

La primera experiencia con FDER fue para las inscripciones a cursos de UCs obligatorias del 2do semestre del 2023. En dicha oportunidad se realizaron 28.918 inscripciones de 7.761 estudiantes, de los cuales 2.929 participaron con prioridad de domicilio. El resultado de la asignación de cupos determinó que 28.802 inscripciones (99,6 %) obtuvieran un cupo en uno de los horarios de su preferencia, en particular 22.809 inscripciones (79 %) fueron asignadas al horario de mayor preferencia estudiantil. Además, el 89,7 % de los estudiantes que presentaron alguna prioridad obtuvieron el horario de su mayor preferencia. Las 116 (0,4 %) inscripciones que no obtuvieron un horario para los cursos, corresponden a 107 estudiantes que quedan en lista de espera y son registrados en el sistema como suplentes en dichos cursos.

En el caso de PSICO, en la primera experiencia realizada para las inscripciones a cursos de UCs del 2do semestre de 2024, se consideraron 26.996 inscripciones de 6.436 estudiantes distintos. Del total de inscripciones 23.735 (87,9 %) quedaron con horario asignado obteniendo un cupo y 3.261 quedaron como suplentes. De las inscripciones que quedaron con horarios asignados, 20.359 (85,8 %) fueron asignadas al horario indicado de mayor preferencia.

En ambos servicios se contó con la posibilidad de que la bedelía realice la gestión de los suplentes posteriormente, algo que no era posible con la metodología anterior para la asignación de cupos. La información recabada por el sistema para alimentar los modelos matemáticos ha permitido adicionalmente tener datos de la demanda estudiantil y con el tiempo poder redefinir horarios, ajustar los criterios de selección y priorización y redefinir modalidades de dictado según demanda (a distancia, presencial, híbrido). Además, es importante destacar que luego de la aplicación de la nueva solución, disminuyeron los picos de acceso durante los períodos de inscripción, lo que ha redundado en una mejora en la disponibilidad del sistema.

Agradecimientos A todos los integrantes de SeCIU que de alguna manera u otra participaron en el proyecto.

Referencias

1. CDC-Udelar, Resolución 6 del 26/11/2021, expediente 001000-501086-21, disponible en URL, último acceso: 2025/11/08

2. Pentico, D.W.: Assignment problems: A golden anniversary survey. European Journal of Operational Research **176**(2), 774–793 (2007)
3. Gale D. and L. Shapley. College admissions and the stability of marriage. American Mathematical Monthly 69 (1962), 9–15.
4. Proll, L.G.: A simple method of assigning projects to students. Journal of the Operational Research Society **23** (2), 195–201 (1972)
5. Chiarandini, M., Fagerberg, R., Gualandi, S.: Handling preferences in student-project allocation. Annals of Operations Research **275** (1), 39–78 (2019)
6. Olaosebikan, S., Manlove, D: Super-stability in the student-project allocation problem with ties. Journal of Combinatorial Optimization **43** (5), 1203–1239 (2022)
7. SGAE-Udelar, Sistema de Gestión Administrativa de Enseñanza, disponible en URL, último acceso: 2025/11/08

Algoritmo evolutivo y redes generativas antagónicas para mejorar la clasificación de conjuntos de datos parcialmente etiquetados

Franco Ferrari[1], Sergio Nesmachnow[1], Jamal Toutouh[2]

[1] Universidad de la República, Uruguay {franco.ferrari,sergion}@fing.edu.uy
[2] Universidad de Málaga, España jamal@uma.es

Abstract. Este artículo presenta un enfoque evolutivo multi-objetivo para entrenar redes generativas antagónicas en escenarios semisupervisados con datos etiquetados extremadamente limitados. El enfoque propuesto trabaja con dos poblaciones independientes: una de generadores y otra de discriminadores, que evolucionan de forma paralela. Los resultados demuestran que el enfoque evolutivo obtiene resultados de precisión superiores y con menor variabilidad, al mismo tiempo que mantiene buenos valores de calidad en las imágenes generadas.

Keywords: Redes generativas antagónicas · algoritmos evolutivos · aprendizaje semisupervisado

1 Introducción

Las redes generativas antagónicas (GANs) han demostrado efectividad para clasificación y generación de imágenes desde su creación en 2014 [2]. Sin embargo, su entrenamiento presenta limitaciones con escasos datos etiquetados. El entrenamiento semisupervisado de GANs con datos etiquetados limitados y una mayor proporción de datos sin etiquetar suele presentar inestabilidad, discriminadores de baja precisión y generadores que colapsan. Este artículo aborda estos problemas mediante hibridación de GANs con algoritmos evolutivos (AE), una estrategia que ha mostrado resultados prometedores en trabajos previos [4–6].

A diferencia del enfoque tradicional con un único par generador-discriminador, este proyecto utiliza un AE multi-objetivo (MOEA) con dos poblaciones independientes, una de generadores y otra de discriminadores, que evolucionan mediante operadores evolutivos y se emparejan dinámicamente durante el entrenamiento.

El MOEA tiene como objetivo mejorar el desempeño de clasificación de los discriminadores en contextos semisupervisados con escasos datos etiquetados. Los resultados experimentales, obtenidos utilizando el dataset MNIST, demuestran que respecto al enfoque no evolutivo, el MOEA logró mejorar significativamente la estabilidad del entrenamiento, logrando precisiones superiores y con menor variabilidad, sin degradar la calidad de las imágenes generadas.

2 Arquitectura base de la GAN

Se diseñó una arquitectura de GAN optimizada para aprendizaje semisupervisado que se híbrido con AEs. La arquitectura utiliza redes neuronales convolucionales, técnicas del estado del arte (batch normalization, dropout, leaky ReLU, label smoothing) y estrategias específicas de distribución de datos etiquetados.

El entrenamiento consideró tres tipos de datos: imágenes reales etiquetadas (K clases), imágenes reales sin etiquetar e imágenes generadas. La salida del discriminador se compone de $K+1$ probabilidades: K para clases reales y una para imágenes generadas. Esta estructura de la salida permite utilizar una función de loss para el discriminador que combina un componente supervisado, que evalúa la clasificación de datos etiquetados en sus clases específicas, y un componente no supervisado que incluye la clasificación de datos no etiquetados como reales y datos generados como falsos [3]. Por otra parte, el generador utiliza dos losses parciales inspirados en índices de calidad y diversidad [6].

3 Hibridación con algoritmos evolutivos

El MOEA propuesto trabaja con dos poblaciones en simultáneo (generadores y discriminadores), cada una inicializada con P individuos. Cada población busca minimizar dos funciones objetivo inspiradas en los losses utilizados en el entrenamiento: los discriminadores optimizan un objetivo supervisado y otro no supervisado, mientras que los generadores optimizan calidad y diversidad.

El proceso evolutivo se organiza en generaciones de ambas poblaciones, las cuales se generan a partir de los operadores de selección, cruzamiento y reemplazo. La selección extrae K individuos padres de cada población mediante un torneo modificado que utiliza non dominated sorting genetic algorithm (NSGA-II) [1]. En el cruzamiento, los padres son clonados generando K descendientes por población, los cuales se emparejan aleatoriamente resultando en K parejas de generador-discriminador. Estas parejas de descendientes son entrenadas durante un número fijo de épocas, durante las cuales modifican sus parámetros y dejan de ser copias idénticas de sus padres. Finalmente, se utiliza NSGA-II como algoritmo de reemplazo, reduciendo cada población ampliada de $P+K$ individuos (padres y descendientes) a P individuos. Este ciclo se repite durante una cantidad fija de generaciones, lo cual permite que cada red enfrente múltiples adversarios, fomentando la exploración y evitando estancamientos.

4 Evaluación experimental

La evaluación se realizó sobre el dataset de dígitos manuscritos MNIST con 60, 100, 500 y 1000 datos etiquetados (de 60000 disponibles) y el resto de los datos como no etiquetados. Se realizaron 30 ejecuciones de 100 épocas para cada enfoque (no evolutivo y MOEA), sobre las cuales se evaluaron la precisión del discriminador y la distancia de inicio de Fréchet (FID) para la calidad de las imágenes generadas.

Los resultados de precisión se presentan en la Figura 1. Con respecto al enfoque no evolutivo, el MOEA incrementó las medias 10% en todos los escenarios y las medianas en hasta 5.43%. Asimismo, redujo los desvíos entre 56% y 97%. Este resultado implica que el MOEA proporciona un entrenamiento más robusto, con resultados replicables. Además, utilizar más individuos permitió mejorar las medias de precisión, un resultado que demuestra que la diversidad poblacional contribuye positivamente al entrenamiento.

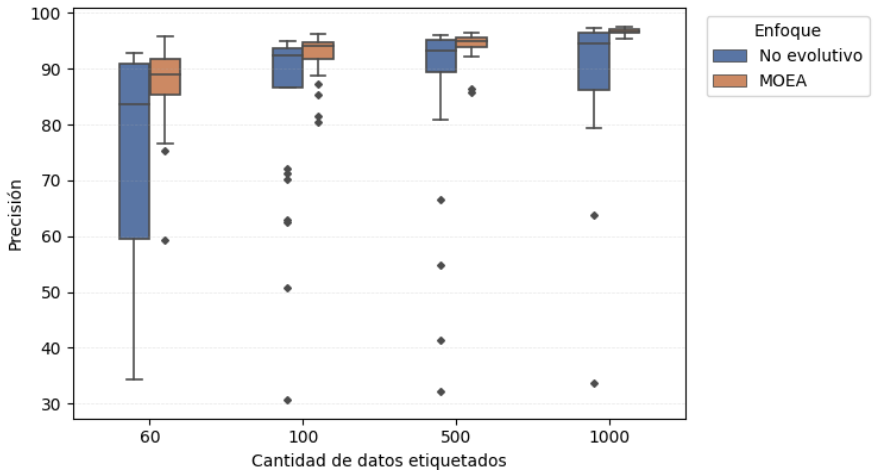


Fig. 1: Boxplots de precisiones por escenario y enfoque. Se omiten valores atípicos extremos (<25%) para el enfoque no evolutivo en 100, 500 y 1000 datos.

Respecto a la calidad de las imágenes generadas, el MOEA obtuvo resultados de FID similares al enfoque no evolutivo, con medias de aproximadamente 10 y desvíos reducidos, sin diferencias significativas en los tests estadísticos realizados.

5 Conclusiones y trabajo futuro

El trabajo realizado demostró que es posible entrenar GANs con datos etiquetados extremadamente limitados y obtener resultados competitivos. La incorporación del MOEA mejoró significativamente la estabilidad del entrenamiento. El MOEA obtuvo precisiones superiores y con menor variabilidad respecto al enfoque no evolutivo, sin degradar la calidad de las imágenes generadas.

Como trabajo futuro se propone integrar el enfoque evolutivo con arquitecturas del estado del arte para verificar mejoras en contextos con resultados superiores. Asimismo, se plantea extender la metodología a datasets más complejos. Finalmente, se sugiere experimentar modificando o agregando nuevas funciones objetivo que puedan mejorar el desempeño del MOEA.

Referencias

1. Deb, K., Agrawal, S., Pratap, A., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans. Evol. Comput. **6**(2), 182–197 (2002)
2. Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A.C., Bengio, Y.: Generative adversarial networks. CoRR **abs/1406.2661** (2014)
3. Salimans, T., Goodfellow, I.J., Zaremba, W., Cheung, V., Radford, A., Chen, X.: Improved techniques for training gans. In: Lee, D.D., Sugiyama, M., von Luxburg, U., Guyon, I., Garnett, R. (eds.) Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain. pp. 2226–2234 (2016)
4. Sedeño, F., Toutouh, J., Chicano, F.: Generate more than one child in your co-evolutionary semi-supervised learning GAN. CoRR **abs/2504.20560** (2025)
5. Toutouh, J., Nalluru, S., Hemberg, E., O'Reilly, U.M.: Semi-supervised learning with coevolutionary generative adversarial networks. In: Proceedings of the Genetic and Evolutionary Computation Conference. pp. 568–576 (2023)
6. Wang, C., Xu, C., Yao, X., Tao, D.: Evolutionary generative adversarial networks. CoRR **abs/1803.00657** (2018)

Grandes Modelos Multimodales para percepción y predicción de metas en robótica

Joel Cabrera Dechia[1] and Guillermo Trinidad Barnech[1]

Facultad de Ingeniería, Universidad de la República, Montevideo, Uruguay
{joel.cabrera, gtrinidad}@fing.edu.uy

Resumen Este trabajo explora el uso de Grandes Modelos Multimodales (LMMs) como herramienta complementaria en la percepción y planificación robótica. A diferencia de los detectores tradicionales, los LMMs aportan descripciones semánticas útiles para la manipulación. Se evaluaron Gemini, ChatGPT-4o y Claude en tareas de detección visual y predicción de metas. Los resultados muestran que estos modelos pueden integrarse eficazmente para conectar lenguaje natural con planificación estructurada, aunque con limitaciones en tiempo de respuesta.

Keywords: modelos multimodales · robótica · detección de objetos · planificación de tareas

1. Introducción

El uso de los LMMs ha crecido notablemente en los últimos años, con modelos conversacionales como ChatGPT o Gemini aplicados en áreas muy diversas [1,2,3]. En robótica, se han empleado como módulos de razonamiento para interpretar lenguaje natural y analizar escenas visuales [4].

Dentro de este contexto, dos aspectos resultan especialmente relevantes: la detección de objetos y la planificación de tareas. La detección de objetos permite localizar e identificar elementos dentro de una imagen [10,11,12], pero no proporciona información sobre cómo manipularlos, algo esencial para la interacción robótica. Por su parte, la planificación de tareas consiste en determinar la secuencia de acciones necesaria para alcanzar un objetivo a partir del estado del mundo [6], siendo fundamental en la interacción humano-robot [7].

El objetivo de este trabajo es evaluar el uso de estos modelos como herramienta complementaria en percepción y planificación robótica, analizando su desempeño en detección con atributos semánticos y en la generación estructurada de metas a partir de lenguaje natural, así como sus principales ventajas y limitaciones.

2. Metodología

2.1. Detección de objetos

En esta prueba, los modelos debían detectar los objetos presentes en una imagen y devolver un resultado en formato JSON describiendo, para cada uno,

nombre, forma, color, tipo de aristas, tamaño, orientación, posición y descripción relativa. Se utilizaron los modelos Gemini Pro 1.5, Gemini Flash 1.5, ChatGPT-4o y Claude.

2.2. Predictor de metas

Se implementaron dos métodos para evaluar la capacidad de Gemini como predictor de metas en planificación robótica, inspirados en [8].

El primer método, **discusión entre modelos**, utiliza dos módulos y puede haber hasta cuatro iteraciones:

- **GoalGenerator (Gemini Flash 1.5)**: genera metas en formato Planning Domain Definition Language (PDDL), un lenguaje estandarizado para describir dominios y problemas de planificación automática [9], junto con sus justificaciones.
  Ejemplo: ante "hacer una ensalada de fruta en la mesa 1", produce `(:goal (and (on banana table1) (on blueberry table1)))`.
- **GoalEvaluator (Gemini Pro 1.5)**: valida las metas generadas y puede solicitar aclaraciones. *Ejemplo:* pregunta si existe un recipiente en la mesa antes de aprobar la meta.

El segundo método, **traducción de instrucciones**, adopta un enfoque más directo y sin iteraciones:

- **Query2Instructions (Gemini Flash 1.5)**: genera pasos en lenguaje natural a partir de una instrucción. *Ejemplo:* para "usar la mesa 2 para trabajar", genera "mover la taza naranja y la naranja desde la mesa 2 a la mesa 1".
- **Instructions2Goals (Gemini Pro 1.5)**: traduce las instrucciones a metas PDDL. *Ejemplo:* genera `(:goal (and (on orange_cup table1) (on orange table1)))`.

3. Resultados y discusión

3.1. Detección de objetos

De manera general, todos los modelos respetaron el formato y coincidieron en las formas detectadas. Gemini Pro presentó mayor precisión en los colores, mientras que Flash fue el más rápido (4 s frente a 9 s). Claude mostró algunas confusiones cromáticas (azul puro vs. verdoso) y ChatGPT-4o ofreció resultados coherentes.

Gemini Pro 1.5 fue el de mejor desempeño, con una precisión promedio de 0.857 ($\pm$0.335) y un tiempo medio de detección de 6.06 s ($\pm$1.28 s). Las variables *edges* y *shape* resultaron las más estables, mientras que *color* fue la más inestable. *Orientation* y *bounding box* mantuvieron un buen equilibrio entre precisión y estabilidad, y *name* mostró mayor dispersión por confusiones entre objetos similares (por ejemplo, manzana vs. naranja o *mug* vs. *cup*).

3.2. Predictor de metas

Discusión entre modelos. El primer método permitió manejar consultas ambiguas; aunque en algunos casos realizaba preguntas innecesarias, lo cual afectaba la estabilidad y el tiempo de respuesta. En esta misma línea, el módulo evaluador tendió a desconfiar del módulo generador, llegando incluso a cuestionar si el robot poseía un brazo, aun cuando esto se había especificado en el *prompt*.

De todas formas, el método fue satisfactorio para consultas poco definidas. Por ejemplo, ante la instrucción "*Quiero trabajar en la mesa 2*", con la mesa ocupada, el módulo generador inicialmente no propuso metas válidas, pero el evaluador sugirió liberar la superficie, permitiendo corregir la respuesta y producir las metas adecuadas.

Traducción de instrucciones. El segundo método mostró un rendimiento más estable y directo. Los módulos trabajaron de forma complementaria: uno generaba los pasos en lenguaje natural y el otro los traducía a metas formales, reduciendo los errores de interpretación y evitando el sobreanálisis.

Por ejemplo, ante "*Quiero hacer una ensalada de fruta en la mesa 3*", el sistema generó correctamente una secuencia de acciones ("mover el plátano y el arándano a la mesa 3") y sus correspondientes metas en PDDL. Además, el tiempo de respuesta fue menor y las respuestas más consistentes entre ejecuciones.

Si bien los resultados son alentadores, el enfoque propuesto presenta limitaciones para escenarios dinámicos y aplicaciones con requerimientos de tiempo real. Los tiempos de inferencia de los LMMs, del orden de varios segundos por consulta, junto con su variabilidad al tratarse de modelos ejecutados sobre servicios remotos, dificultan su integración en lazos de control reactivos.

4. Conclusiones y trabajo futuro

Los resultados obtenidos indican que los LMMs pueden desempeñar un papel complementario en sistemas robóticos, aportando información semántica y geométrica útil para la manipulación de objetos, más allá de la simple detección.

En el predictor de metas, el enfoque basado en discusión entre modelos permitió manejar consultas ambiguas, aunque tendió al sobreanálisis y a la generación de predicados innecesarios. El método de traducción de instrucciones, en cambio, mostró mayor estabilidad y velocidad, produciendo metas correctas en todos los casos evaluados. Como mejora futura, se propone combinar ambos enfoques para aprovechar sus ventajas.

Como trabajo futuro, se plantea integrar estos módulos con sistemas perceptivos y de planificación clásicos, así como reducir la latencia para su posible uso en escenarios más dinámicos.

Referencias

1. Dong, Y., Jiang, X., Jin, Z., Li, G.: Self-Collaboration Code Generation via ChatGPT. ACM Trans. Softw. Eng. Methodol. **33**(7), 189 (2024). https://doi.org/10.1145/3672459
2. Valova, I., Mladenova, T., Kanev, G.: Students' Perception of ChatGPT Usage in Education. Int. J. Adv. Comput. Sci. Appl. **15**(1) (2024)
3. Shahsavar, Y., Choudhury, A.: User Intentions to Use ChatGPT for Self-Diagnosis and Health-Related Purposes: Cross-sectional Survey Study. JMIR Hum. Factors **10**, e47564 (2023). https://doi.org/10.2196/47564
4. Ahn, M., Brohan, A., Brown, N., et al.: Do As I Can, Not As I Say: Grounding Language in Robotic Affordances. arXiv:2204.01691 [cs.RO] (2022)
5. Xu, G., Khan, A., Moshayedi, A., et al.: The Object Detection, Perspective and Obstacles in Robotic: A Review. EAI Endorsed Trans. AI Robot. **1**(1), e13 (2022)
6. Zhao, Z., Cheng, S., Ding, Y., et al.: A Survey of Optimization-Based Task and Motion Planning: From Classical to Learning Approaches. IEEE/ASME Trans. Mechatronics **30**(4), 2799–2825 (2025). https://doi.org/10.1109/TMECH.2024.3452509
7. Tellex, S., Kollar, T., Dickerson, S., et al.: Understanding Natural Language Commands for Robotic Navigation and Mobile Manipulation. Proc. AAAI Conf. Artif. Intell. **25**(1), 1507–1514 (2011)
8. Kenton, Z., Siegel, N., Kramár, J., et al.: On Scalable Oversight with Weak LLMs Judging Strong LLMs. arXiv:2407.04622 [cs.LG] (2024)
9. D. McDermott, M. Ghallab, A. Howe, C. Knoblock, A. Ram, M. Veloso, D. Weld, and D. Wilkins, *PDDL — The Planning Domain Definition Language*. Technical Report CVC TR-98-003, Yale Center for Computational Vision and Control, 1998.
10. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You Only Look Once: Unified, Real-Time Object Detection. In: Proc. CVPR, pp. 1–10 (2016)
11. Liu, W., Anguelov, D., Erhan, D., et al.: SSD: Single Shot MultiBox Detector. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016, pp. 21–37. Springer, Cham (2016)
12. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. arXiv:1506.01497 [cs.CV] (2016)
13. Lin, T.-Y., Goyal, P., Girshick, R., et al.: Focal Loss for Dense Object Detection. arXiv:1708.02002 [cs.CV] (2018)

Redes neuronales generativas antagónicas para mejorar la calidad de sistemas de identificación facial

Ana Paula Mazzeo, María Agustina Mazzeo, Sergio Nesmachnow

Universidad de la República, Uruguay
{ana.paula.mazzeo,maria.agustina.mazzeo,sergion}@fing.edu.uy

Abstract. Este artículo presenta un enfoque evolutivo multiobjetivo para la generación de rostros sintéticos a partir del espacio latente de una red generativa antagónica. Se integra un algoritmo evolutivo con dos evaluadores de similitud facial y características de raza. Las soluciones obtenidas permiten generar imágenes que combinan similitud con un rostro objetivo y correspondencia con una raza predefinida.

Keywords: Reconocimiento facial · GANs · Algoritmos evolutivos

1 Introducción

El reconocimiento facial se ha convertido en una tecnología central en ámbitos como la seguridad, la autenticación biométrica y tareas cotidianas, como el desbloqueo de dispositivos móviles [3]. En paralelo, la generación de rostros sintéticos permite estudiar sesgos demográficos, construir conjuntos de datos más equilibrados y explorar escenarios difíciles de obtener con datos reales, siendo las GANs especialmente eficaces para producir imágenes faciales de alta calidad [2]. Sin embargo, controlar con precisión los atributos generados sigue siendo un desafío debido a la alta dimensionalidad del espacio latente. En este contexto, los algoritmos evolutivos ofrecen una estrategia adecuada para explorar el espacio latente, mantener diversidad en las soluciones y optimizar múltiples objetivos simultáneamente.

Este trabajo propone la implementación de un algoritmo evolutivo multiobjetivo (MOEA) con modelos generativos, de reconocimiento facial y de características para la generación de rostros sintéticos con rasgos controlados.

La combinación de los componentes de similitud facial y de raza busca evaluar la utilidad de las imágenes generadas para construir conjuntos de datos representativos para todas las categorías raciales. Los conjuntos de datos pueden utilizarse para estudiar el comportamiento de los sistemas de reconocimiento facial ante variaciones sistemáticas en los atributos faciales.

Los resultados obtenidos evidencian que el MOEA propuesto es capaz de generar rostros sintéticos que cumplen con los objetivos de similitud facial y clasificación racial. Además, el uso de más de un reconocedor facial contribuyó a una evaluación más robusta, reduciendo el acoplamiento a un único modelo.

2 Arquitectura de la solución

La solución integra un algoritmo evolutivo multiobjetivo con módulos de generación, reconocimiento y clasificación racial. El sistema busca optimizar la generación de rostros sintéticos que presenten simultáneamente similitud con una imagen objetivo y que correspondan a una raza predefinida.

Los módulos seleccionados para la solución incluyen, para la GAN, StyleGAN3, elegido por su invarianza a la traslación y rotación, lo que evita las deformaciones presentes en StyleGAN2 y facilita la exploración del espacio latente [2]. Para el reconocimiento facial se emplearon VGG-Face y Dlib, debido a su precisión superior a la humana en LFW [4]. También se incorporó FairFace, entrenado con un conjunto de datos demográficamente equilibrado para mejorar la clasificación de raza [1]. Finalmente, el algoritmo evolutivo utilizado es NSGA-II por su eficiencia en la optimización multiobjetivo y su capacidad para mantener diversidad en la población.

El proceso completo puede describirse en tres fases principales:

1. Inicialización: se genera una población inicial de vectores aleatorios a partir de una distribución normal estándar. Cada vector representa un punto en el espacio latente de StyleGAN3.
2. Evaluación: cada vector se procesa mediante la GAN para producir una imagen facial sintética. La imagen se evalúa con VGG-Face y Dlib para determinar la similitud con la imagen objetivo y con FairFace para obtener la clasificación de raza. La similitud y la clasificación de raza conforman las funciones objetivo.
3. Optimización evolutiva: los valores de las funciones objetivo se emplean en un MOEA, que aplica operadores de selección, cruzamiento y mutación sobre los vectores latentes para generar la nueva población. El proceso se repite durante un número fijo de generaciones hasta obtener un conjunto de soluciones que optimice simultáneamente ambos objetivos.

3 Validación experimental

La evaluación se realizó sobre nueve instancias en total, utilizando treinta ejecuciones independientes del algoritmo para cada una. La configuración empleada incluyó una población de 80 individuos y un máximo de 100 generaciones. El cruzamiento se implementó mediante el operador *blend-*α, con $\alpha = 0.2$ y una probabilidad de 0.9. La mutación gaussiana se aplicó con una probabilidad de 0.001, y la selección de padres se realizó mediante torneos binarios.

En la Fig. 1 se presentan las imágenes correspondientes a casos representativos de las soluciones finales generadas por el algoritmo evolutivo para la instancia I7. Para analizar la tasa de acierto, se midió la proporción de imágenes generadas que fueron identificadas correctamente como el individuo de referencia y la proporción que coincidió con la raza objetivo. En la Tabla 1 se presentan las tasas de acierto para todas las instancias.

Fig. 1: Instancia I7 y ejemplos de soluciones generadas por el algoritmo.

Table 1: Tasa de aciertos por instancia

Instancia	VGG-Face	Dlib	FairFace
I1	72.6%	77.2%	79.2%
I2	89.0%	97.3%	90.3%
I3	81.4%	77.3%	72.2%
I4	94.3%	95.1%	97.2%
I5	69.9%	78.2%	69.4%
I6	44.9%	70.6%	90.3%
I7	63.3%	78.3%	97.0%
I8	87.2%	66.7%	82.5%
I9	35.9%	63.4%	77.2%

4 Conclusiones y trabajo futuro

Los resultados mostraron que el algoritmo evolutivo es capaz de generar soluciones que satisfacen simultáneamente ambos objetivos en la mayoría de las instancias evaluadas. En particular, se identificó un desempeño consistente en la clasificación de raza, con altos porcentajes de coincidencia con la raza objetivo, mientras que la similitud con el individuo objetivo presentó mayor variabilidad.

Como líneas de trabajo futuro, resulta relevante extender el análisis mediante la incorporación de un mayor número de reconocedores o hacia otros atributos, como la edad, el género o la expresión facial, con el fin de estudiar cómo interactúan múltiples características en la generación de rostros sintéticos. Además, un camino prometedor consiste en emplear las imágenes generadas para entrenar o reforzar sistemas de reconocimiento facial, evaluando si estos conjuntos sintéticos pueden contribuir a reducir sesgos y mejorar la robustez de los sistemas actuales.

Referencias

1. Karkkainen, K., Joo, J.: Fairface: Face attribute dataset for balanced race, gender, and age for bias measurement and mitigation. In: CVF Winter Conference on Applications of Computer Vision (2021)
2. Karras, T., Aittala, M., Laine, S., Härkönen, E., Hellsten, J., Lehtinen, J., Aila, T.: Alias-free generative adversarial networks. In: Proc. NeurIPS (2021)
3. Mohsienuddin, S., Sabri, M.: Facial recognition technology. SSRN Electronic Journal **7**, 176–184 (06 2020)
4. Serengil, S.I., Ozpinar, A.: A benchmark of facial recognition pipelines and co-usability performances of modules. Bilisim Teknolojileri Dergisi (2024)

Redes neuronales basadas en operadores neuronales profundos para la construcción de modelos subrogados aplicados a la detección de daño por heladas agrometeorológicas

Lucas González Petti, Ángel Retali, Sergio Nesmachnow

Universidad de la República, Uruguay
lucas.gonzalezpetti@outlook.com,retisito@gmail.com,sergion@fing.edu.uy

Abstract. Este artículo presenta un enfoque basado en redes neuronales con operadores neuronales profundos para el problema de construir modelos subrogados aplicados a la detección de daño por heladas agrometeorológicas

Keywords: Redes neuronales· operadores profundos · modelos subrogados · agrometeorología · daño por heladas agrometeorológicas

1 Introducción

Las heladas agrometeorológicas amenazan la producción agrícola, especialmente en regiones donde la topografía favorece la acumulación de aire frío. La identificación de zonas de riesgo y la evaluación de estrategias de mitigación requiere modelos numéricos capaces de reproducir con precisión la dinámica térmica nocturna en el entorno de los cultivos. En este contexto, las simulaciones de mecánica de los fluidos computacional (CFD) se han consolidado para estudiar la interacción entre topografía, estratificación térmica y forzantes atmosféricas [3].

A pesar de su utilidad, las simulaciones CFD aplicadas a dominios extensos presentan desafíos significativos debido a los elevados tiempos y recursos de cómputo requeridos. Las simulaciones actuales permiten cubrir dominios del orden de 20×20 km pero con demandas de 400 GB de memoria GPU y la utilización prolongada de hardware especializado, con costos del orden de 500 USD por simulación. Estas exigencias restringen la posibilidad de abordar regiones de mayor extensión o de incorporar el modelado a sectores productivos más amplios, como el forestal, que demandaría recursos computacionales aún más importantes.

Ante estas limitaciones, las técnicas de aprendizaje profundo se han explorado para construir modelos subrogados capaces de reproducir la física relevante con un costo computacional significativamente menor. En particular, las redes neuronales de operadores profundos (DeepONets) han mostrado un gran potencial al aproximar operadores lineales y no lineales de manera eficiente. Su arquitectura basada en redes branch y trunk permite capturar relaciones funcionales complejas y su combinación con enfoques informados por física (PINNs) ha demostrado resultados altamente prometedores en diversas áreas de aplicación [2, 4].

Este trabajo propone aplicar operadores neuronales profundos al problema de detección de zonas de riesgo de heladas. Se trabaja con simulaciones para estimar el campo de temperaturas al final de la noche bajo un forzante de enfriamiento radiativo que actúa diferencialmente en altura. Se explora una arquitectura de DeepONet que se valida experimentalmente sobre un escenario real del problema de detección de daño por heladas agrometeorológicas. Los resultados preliminares muestran valores correctos de precisión para el caso de estudio abordado. El desarrollo de modelos subrogados basados en aprendizaje profundo permitirá extender el análisis a dominios más amplios y evaluar estrategias de mitigación con una reducción sustancial de los costos de cálculo.

2 Metodología y arquitectura base de la DeepONet

El objetivo del problema es predecir el campo de temperatura sobre un terreno a partir de un conjunto de datos de entrada que incluyen la información topográfica (alturas) y valores de base (ground truth) estimaciones calculadas mediante métodos numéricos de CFD en las zonas de estudio.

La metodología aplicada se basa en plantear el problema como un aprendizaje supervisado usando redes neuronales con la arquitecturas DeepONet. Las DeepONets proponen un enfoque basado en redes neuronales para aproximar operadores no lineales entre espacios funcionales. En lugar de modelar directamente una función entre variables, DeepONet busca aprender un mapeo que asocie funciones de entrada con funciones de salida, i.e., un operador $\mathcal{G} : u \mapsto \mathcal{G}u = v$, con u y v funciones definidas sobre dominios continuos. Esta capacidad es especialmente útil en contextos donde se necesita emular soluciones de ecuaciones diferenciales, sistemas físicos o modelos computacionales costosos de evaluar.

La arquitectura DeepONet se compone de dos redes: la *branch net* y la *trunk net*. La trunk net recibe como entrada las coordenadas espaciales (o temporales) en las que se desea evaluar la función de salida v y produce un vector latente que representa la dependencia funcional respecto del dominio espacial (o temporal). Por otro lado, la branch net toma como entrada una muestra discreta de la función de entrada u, evaluada en un conjunto fijo de puntos, y produce también una representación latente de la entrada. La salida final de la DeepONet se obtiene mediante un producto interno entre los vectores producidos por ambas redes, permitiendo así que la predicción final dependa tanto de la función de entrada como el punto específico de la evaluación.

En el caso de estudio, la arquitectura DeepONet se adaptó para modelar las relaciones funcionales entre las características físicas del espacio de estudio y las distribuciones de temperatura, para generalizar nuevas regiones con topografía desconocida. Dentro de las variantes propuestas en la literatura, la arquitectura DeepONet2D [1] se adecúa específicamente a situaciones donde las funciones de entrada y salida están definidas sobre dominios bidimensionales. En este caso, la branch net recibe como entrada campos discretizados en dos dimensiones (por ejemplo, mapas o imágenes), lo que permite capturar patrones espaciales mediante capas convolucionales u otras arquitecturas diseñadas para datos 2D.

De manera complementaria, la trunk net utiliza coordenadas bidimensionales (x, y) como entrada y genera representaciones latentes adaptadas a la variación espacial dentro del dominio. Esta formulación facilita la evaluación punto a punto en mallas continuas y habilita la interpolación y extrapolación espacial con mayor precisión que la arquitectura unidimensional estándar.

3 Análisis exploratorio de datos

El conjunto de datos original comprende 1303 archivos distribuidos en 89 directorios con estructuras FL y GR, y contiene campos tridimensionales y cuatridimensionales de hasta 512×512×88×3. El tamaño de cada archivo es mayor a 2.8 GB. Esta magnitud dificulta su utilización directa para tareas de análisis o entrenamiento. Con el fin de hacer viable el procesamiento, se implementó una estrategia de reducción que consistió en conservar únicamente las primeras 20 capas en profundidad. Posteriormente, sobre estos archivos reducidos, se efectuó un análisis exploratorio enfocado en variables críticas: presión, temperatura, viscosidad, volumen de celda y las componentes verticales de altura.

El cálculo de estadísticas descriptivas a partir de estas variables permitió generar un dataset tabular compacto, adecuado para comparaciones y análisis subsiguientes. La matriz de correlación resultante se presenta en la Fig. 1 y revela relaciones físicas consistentes, tales como la correlación positiva entre volumen y altura, la relación negativa entre presión y temperatura, y una correlación moderada entre temperatura y altura. Estos resultados confirman la coherencia del dataset procesado y su idoneidad para futuras tareas de modelado y aprendizaje automático.

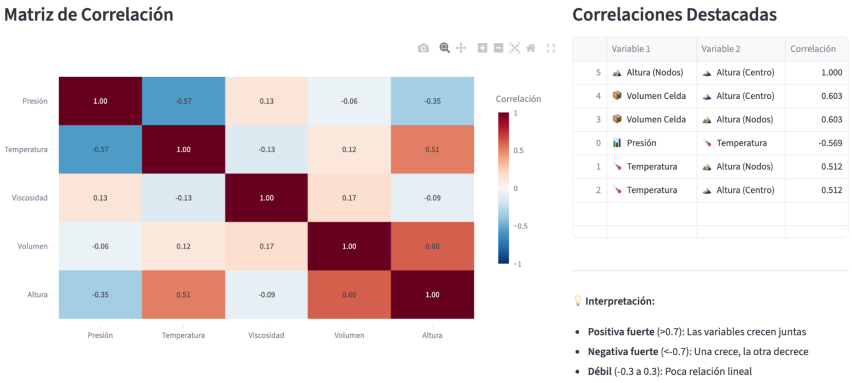


Fig. 1: Matriz de correlación determinada en el análisis exploratorio de datos

4 Análisis experimental

El modelo desarrollado se entrenó utilizando datos preprocesados derivados de los archivos originales mencionados en la Sección 3. Si bien se cuenta con 20 capas en profundidad de la malla, el objetivo es predecir la temperatura a un metro del suelo. La capa que mejor se aproxima a esta medida es la número 5, por lo tanto Los experimentos consideraron esta capa para el entrenamiento. y un rango de 2 a 5 capas superiores e inferiores a la objetivo. Los mejores resultados de inferencia se obtuvieron utilizando dos capas.

Sobre las estructuras de FL y GR se aplicó un esquema de muestreo basado en parches bidimensionales, con el objetivo de capturar tanto información local como variaciones topográficas y físicas relevantes. Para asegurar una cobertura espacial adecuada y aumentar la diversidad del conjunto de entrenamiento, se emplearon ventanas deslizantes con solapamiento controlado, que permitieron generar múltiples instancias por archivo sin incurrir en redundancia excesiva. En particular, se tomaron parches representativos a 1km^2 en ventanas deslizantes con un solapamiento del 75% entre parches.

Durante los experimentos se consideraron distintas variables físicas en la branch net y para la trunk net siempre se utilizó la variable posicional de la altura del terreno z. Para el experimento que obtuvo mejores resultados se utilizó la arquitectura DeepONet2D y la variable Presión (P) en la branch net. Para el conjunto de test se obtuvo un RMSE de 0.226°C, una relación con la dispersión de T de 0.213σ y una relación con su rango máximo y mínimo de 2.12%.

5 Conclusiones y trabajo futuro

El trabajo realizado ha mostrado la capacidad de las DeepONets para la construcción de modelos subrogados para el problema de detección de daño por heladas agrometeorológicas. Las principales líneas de trabajo actual y futuro se enfocan en ampliar el análisis de datos para reducir el volumen de información a considerar en el entrenamiento de los modelos de DeepONet para reducir el tiempo de cómputo, y ampliar el análisis experimental para involucrar diferentes excenarios y las capacidades de tansferencia de los modelos implementados.

Referencias

1. Bai, H., Wang, Z., Chu, X., Deng, J., Bian, X.: Data-driven modeling of unsteady flow based on deep operator network (2024), https://arxiv.org/abs/2404.06791
2. Lu, L., Jin, P., Karniadakis, G.E.: Deeponet: Learning nonlinear operators for identifying differential equations based on the universal approximation theorem of operators. arXiv preprint arXiv:1910.03193 (2019)
3. Stull, R.: Practical Meteorology: An Algebra-based Survey of Atmospheric Science. University of British Columbia Press (2017)
4. Wang, S., Teng, Y., Perdikaris, P.: Understanding and mitigating gradient pathologies in physics-informed neural networks. SIAM Journal on Scientific Computing **43**(5), A3055–A3081 (2021)

Framework to evaluate sustainable urban mobility: integrating multicriteria analysis, air quality modeling, and traffic microsimulation

Yamila S. Grassi[1-2][0000-0002-9655-4643], Daniel A. Rossit[1-2][0000-0002-2381-4352], Mónica F. Díaz[3-4][0000-0002-6680-8067] and Diego G. Rossit[1-2][0000-0002-8531-445X]

[1] Instituto de Matemática de Bahía Blanca (UNS-CONICET), Bahía Blanca, Argentina.
[2] Departamento de Ingeniería, Universidad Nacional del Sur, Bahía Blanca, Argentina.
[3] Planta Piloto de Ingeniería Química (UNS-CONICET), Bahía Blanca, Argentina.
[4] Departamento de Ingeniería Química, Universidad Nacional del Sur, Bahía Blanca, Argentina.
yamila.grassi@uns.edu.ar

Abstract. This work explores methodological approaches to evaluate the impact of urban mobility on environmental indicators in medium-sized cities. Multicriteria methods are presented to weight sustainability indicators, highlighting the relevance of environmental ones. In local contexts lacking emission, and traffic data, air quality modeling and traffic microsimulation emerge as valuable tools to support evidence-based decision-making and to foster more sustainable urban mobility.

Keywords: Air Quality Modeling, Traffic Microsimulation, Environmental Indicators.

1 Introduction

Sustainable urban mobility is essential for city life, ensuring access to daily activities and goods while meeting travel needs without compromising environmental, health, or economic conditions for future generations [1, 2]. However, it also generates negative impacts such as air pollution, congestion, and traffic accidents [1, 3]. In the context of rapid urban growth, a key challenge is advancing toward sustainable and intelligent transport systems. This requires indicators capable of representing complex phenomena and guiding decision-making, always adapted to the local context under evaluation. In this sense, the present work explores methodological approaches to assess urban mobility impacts on air quality in medium-sized Latin American cities.

2 Proposed methodological framework

Indicators allow us to measure and evaluate complex concepts through clear metrics, enabling comparison of alternatives and guiding decision-making. They are widely used to assess the sustainability of urban mobility. A review of different works

identifies more than 300 potential indicators [4], highlighting the need to carefully select sets of indicators adapted to the local context, since measuring aspects not representative of the region has limited value. An initial selection can be carried out by convening a panel of experts and applying the Delphi method to reach consensus on the most appropriate indicators. Once selected, indicators are typically weighted using multi-criteria decision-making methods, such as the Analytic Hierarchy Process (AHP) or the Best-Worst Method (BWM). Both are based on pairwise comparisons of indicators, but differ in the number of comparisons required: for AHP the number of comparisons is equal to $\frac{n*(n-1)}{2}$, where n is the number of indicators, while BWM requires significantly fewer comparisons, specifically $2n-3$ [5]. Through pairwise comparisons, environmental indicators tend to receive the highest weightings and are often the most frequently used [1, 6]. The assessment of environmental indicators requires data on pollutant and greenhouse gas emissions, as well as air quality associated with urban mobility. This is straightforward when monitoring data are available. However, in contexts where such data are absent, as in medium-sized Latin American cities, air quality models and traffic microsimulation become valuable tools to construct environmental indicators and support evidence-based decision-making.

3 Discussions

Air quality modeling applied to critical points of a city is introduced as a way to estimate traffic-related atmospheric pollution. This methodology requires data on vehicle counts, emission factors, and meteorological variables, which our research group has collected since 2020. This approach enables the identification of areas with higher exposure and establishes links between mobility patterns and environmental risks. In Bahía Blanca, Argentina, air quality modeling has been used to evaluate whether air quality levels in strategic areas with high traffic flow, such as the city center, exceed the limits established by current legal regulations [7]. The analysis also verified whether the modeled concentrations affect public health by comparing them with the categories of the Air Quality Index proposed by the United States Environmental Protection Agency. These results demonstrate the usefulness of air quality modeling and highlight its potential to support the development of air quality indicators based on empirical data. In Bahía Blanca, the modeling is carried out using static emission factors that consider vehicle type, fuel, and emission control technology. However, these factors do not account for driver behavior or vehicle accelerations and decelerations. To address this limitation, traffic microsimulation is introduced as an approach capable of generating more accurate estimates of traffic-related emissions. This tool seeks to overcome the limitations of static emission factors used in traditional air quality models by incorporating variations in speed, flow behavior, and local conditions. The objective is to move toward more precise and adaptable models for complex urban scenarios. Emission data obtained through microsimulation can be used directly in environmental indicators (pollutant and noise emissions). Furthermore, additional indicators can be derived, including those related to the social dimension (accidents) and technical or operational aspects (queue length, travel time). Otković et al. [8] present a comparable

methodology applied in the city of Belišće (Croatia). In parallel, our research group is developing a framework using SUMO software to implement traffic microsimulation at strategic points in Bahía Blanca [9]. Although complete results are not yet available, the methodological has already been piloted in Bahía Blanca. Air quality modeling has enabled the evaluation of downtown areas with high traffic flow, while the microsimulation-based approach is under development to refine emission estimates and strengthen the construction of indicators.

4 Conclusions

In conclusion, the integration of air quality modeling and traffic microsimulation into the construction of environmental indicators provides a robust framework to support evidence-based decision-making. These approaches offer valuable tools for medium-sized Latin American cities to advance toward more sustainable and intelligent urban mobility systems.

References

1. da Silva, R., Santos, G., Setti, D.: A multi-criteria approach for urban mobility project selection in medium-sized cities. Sustainable Cities and Society **86**, 104096 (2022).
2. Vidović, K., Šoštarić, M., Budimir, D.: An overview of indicators and indices used for urban mobility assessment. PROMET - Traffic & Transportation **31**(6), 703-714 (2019).
3. Francois, C., Coulombel, N.: Environmental assessment of daily mobility for the design of neighbourhoods: Coupling mobility microsimulation and life-cycle assessment. Sustainable Cities and Society **105**, 105341 (2024).
4. Grassi, Y., Díaz, M., Rossit, D.: Review of indicators and multi-criteria decision-making methods for assessing the sustainability of urban mobility. Frontiers in Future Transportation, under review dic-2025.
5. Grassi, Y., Díaz, M., Rossit, D.: Overview of indicators and MCDM methods applied to sustainable urban freight transport. In: International Conference on Production Research (ICPR), 2025 (In Press).
6. Mahmoudi, R., Shetab-Boushehri, S., Hejazi, S., Emrouznejad, A.: Determining the relative importance of sustainability evaluation criteria of urban transportation network. Sustainable Cities and Society **47**, 101493 (2019).
7. Grassi, Y., Díaz, M.: Urban air pollution evaluation in downtown streets of a medium-sized Latin American city using AERMOD dispersion model. Environmental Monitoring and Assessment **196**, 521 (2024).
8. Ištoka Otković, I., Karleuša, B., Deluka-Tibljaš, A., Šurdonja, S., Marušić, M.: Combining traffic microsimulation modeling and multi-criteria analysis for sustainable spatial-traffic planning. Land **10**(7), 666 (2021).
9. Rossit, D., Grassi, Y., Díaz, M., Rossit, D.: Propuesta para evaluar indicadores claves de movilidad urbana mediante el uso de microsimulación de tránsito. Caso de estudio: Bahía Blanca, Argentina. In: Congreso Internacional XXXVIII ENDIO – XXXVI EPIO, 2025 (In Press).

Smart Urban Mobility: AI, Quantum Computing and IoT for Sustainable Cities in Latin America

Edwin Gerardo Acuña Acuña[1][0000-0001-7897-4137]

[1] Universidad Latinoamericana de Ciencia y Tecnología, San José, Costa Rica
eacunaa711@ulacit.ed.cr

Abstract. In the twenty-first century, fast digital innovation, increased urbanization, and mounting environmental challenges are driving a key turning point in Latin American urban transportation. Nowadays, around 80% of the region's population lives in cities, creating mobility networks marked by traffic jams, pollution, and excessive energy use. By restricting access to work possibilities for people from different socioeconomic backgrounds, these factors worsen inequality and lower productivity. By combining quantum computing (QC), artificial intelligence (AI), cyber-physical systems (CPS), and the Internet of Things (IoT) to build real-time, adaptive, socio-technical ecosystems, smart-city models have become more relevant in response to these systemic issues [1]. In light of this, mobility solutions need to advance beyond conventional operational paradigms and integrate dynamic intelligence, human-centered governance, and sustainable design [2].

New system designs that are able to forecast, optimize, and learn across numerous layers of information are necessary to address these issues. Urban mobility is still largely dependent on traditional optimization methods, which are unable to handle the combinatorial complexity of dynamic transportation networks, according to [3]. There are three significant flaws in the material now in publication:

(1) AI-based routing models frequently perform poorly in real-time, non-linear, multi-objective scenarios; (2) traditional frameworks hardly ever incorporate digital technologies with explicit environmental performance metrics like energy efficiency and carbon reduction [4]; and (3) the Industry 5.0 paradigm's ethical, participatory, and human-centered principles are not sufficiently integrated, leading to fragmented governance and low social engagement [5]. Data heterogeneity and infrastructure differences among Latin American cities exacerbate these problems, impeding the creation of coherent, robust, and sustainable transportation ecosystems. By developing hybrid AI–QC–IoT mobility architectures that can coordinate adaptable and ecologically conscious transportation networks, our study seeks to close this scientific gap. In addition to technological viability, the goal is to comprehend how sustainability, intelligence, and computing interact to influence urban life in the future.

The paper suggests a multi-layered computational architecture that combines AI-driven predictive analytics, quantum algorithms for large-scale optimization, and IoT-enabled sensing for cyber-physical control. The methodological design follows four stages. First, a theoretical model describes the interactions between AI, QC, and IoT

layers using distributed CPS and digital-twin concepts. Second, demand forecasting is performed using Long Short-Term Memory (LSTM) networks, while quantum-inspired algorithms (QUBO and QAOA) address high-dimensional, multi-objective routing and scheduling problems. Third, an IoT-enabled edge computing layer ensures adaptive connectivity, reduced latency, and lower energy consumption between vehicles, sensors, and infrastructure. Simulations and validations are performed using Python, Qiskit, and SUMO across **pre-existing open mobility datasets from Latin American cities**, including data from Bogotá, São Paulo, and Mexico City (details added per Review 1).

Exploratory simulation results show substantial improvements attributable to the combined effect of AI-based prediction, quantum optimization, and IoT-driven responsiveness. When compared to traditional routing and control methods, the hybrid design decreased CO2 emissions by about 30%, increased network resilience by 15%, and cut average travel durations by up to 40% [7]. These benefits result from the combination of machine learning's predictive power, quantum optimization's combinatorial efficiency, and real-time IoT feedback loops.

According to systems theory, adaptive interactions between computational, physical, and social subsystems lead to sustained mobility. The findings show the significance of edge computing for preserving system responsiveness under large data loads and emphasize the significance of energy-efficient routing methods that strike a compromise between computational and environmental goals. Additional evidence for the necessity of integrated IoT standards in urban ecosystems comes from recent research by Waqar et al. [8]. However, there are still issues, such as the early stages of quantum real-time applications and data interoperability across heterogeneous IoT networks. In addition, it is imperative to address governance elements like privacy protection, transparency, and fair access to avoid smart mobility systems perpetuating current disparities.

Studying makes three contributions. It first develops an adaptable theoretical framework for quantum-enhanced urban intelligence that combines AI, CPS, and sustainability. It also illustrates the technological capability of hybrid computational ecosystems in addressing intricate, non-linear mobility issues. Third, by encouraging cross-sector cooperation, open data standards, and modular infrastructures that facilitate long-term digital transformation, it suggests a paradigm appropriate for Latin American cities. The suggested approach presents smart mobility as an ethical, ecologically conscious, and human-centered system that may enhance quality of life in the framework of Industry 5.0.

Lastly, the design promotes the shift from reactive transportation management to resilient, adaptable, and self-optimizing ecosystems. The model gives Latin America a realistic route to transparent, data-driven, low-carbon transportation by fusing IoT, AI, and quantum computing into a sustainability-focused engineering framework. For Sustainable Smart Cities 5.0, where human purpose and digital intelligence come together

to create the next generation of intelligent, robust, and equitable transportation systems, the suggested paradigm offers a conceptual and practical basis.

Keywords: Artificial Intelligence, Internet of Things, Smart Mobility, Quantum Computing, and Sustainable Cities.

References

1. **Chen, Y., Chan, W. H., Su, E. L. M., & Diao, Q.** (2025). *Multi objective optimization for smart cities: A systematic review*. PeerJ Computer Science, 11, e3042. https://doi.org/10.7717/peerj-cs.3042
2. **Sodhro, A. H., Zongwei, L., Pirbhulal, S., & Sodhro, G. H.** (2023). *Toward machine learning enabled sustainable smart cities*. IEEE Access, 11, 28688–28705. https://doi.org/10.1109/ACCESS.2023.3255147
3. **Bibri, S. E.** (2023). *Advancing urban intelligence with artificial intelligence and digital twins*. Cities, 141, 104594. https://doi.org/10.1016/j.cities.2023.104594
4. **Pallonetto, F., De Rosa, M., & Finn, D.** (2024). *Smart energy systems and flexibility: A comprehensive review*. Renewable and Sustainable Energy Reviews, 187, 113643. https://doi.org/10.1016/j.rser.2023.113643
5. **Gao, S., Ding, W., & Yu, T.** (2024). *AI driven urban mobility optimization: Methods and applications*. Transportation Research Part C, 157, 104470. https://doi.org/10.1016/j.trc.2023.104470
6. **Nahavandi, S.** (2023). *Industry 5.0: A human centric solution*. Sustainability, 15(5), 4093. https://doi.org/10.3390/su15054093
7. **Acuña Acuña, E. G.** (2024). *Healthcare cybersecurity: Data poisoning in the age of AI*. Journal of Comprehensive Business Administration Research.
8. **Waqar, A., Barakat, T. A. H., Almujibah, H. R., Alshehri, A. M., Alyami, H., & Alajmi, M.** (2025). *Analytical approach to smart and sustainable city development with IoT*. Scientific Reports, 15, 23617. https://doi.org/10.1038/s41598-025-08861-y

Operations management in customized production, models and optimization approaches

Daniel Rossit[1,2] and Jeanette Rodriguez[1] and Camila Castellano[1] and Felicitas Marcenac[1] and Tadeo Vega[1] and Diego Rossit[1,2]

[1] Engineering Department, Universidad Nacional del Sur, Bahia Blanca, Argentina
[2] INMABB, CONICET, Bahía Blanca, Argentina
daniel.rossit@uns.edu.ar

Abstract. The transition from mass production to mass customization in Industry 4.0/5.0 environments has introduced new challenges for production planning and scheduling. This presentation examines two complementary approaches to managing operations in personalized production systems. First, we analyze the flow shop scheduling problem with missing operations, where jobs may skip certain stages depending on customer specifications. This variability complicates sequencing and requires balancing multiple objectives such as makespan, total tardiness, and completion time. Multiobjective evolutionary algorithms (MOEAs) including NSGA-II, NSGA-III, MOEA/D, SPEA2, and AGEMOEA-II are shown to provide competitive solutions, enabling resilient and agile scheduling in heterogeneous shop-floor environments. Second, we explore additive manufacturing (AM), a technology that typifies customization by enabling complex geometries and individualized designs. Here, the integrated problem of nesting and scheduling is addressed through hybrid heuristics and mathematical programming. Results demonstrate that heterogeneous builds improve makespan efficiency and drastically reduce computational time, offering practical rules for grouping parts and enhancing industrial-scale AM competitiveness. By analyzing both approaches together, the study highlights how different optimization models confront variability in personalized production. The findings provide actionable insights for improving efficiency, resilience, and adaptability in modern manufacturing systems.

Keywords: Mass Customizatoin, Flow Shop Scheduling, Additive Manufacturing, Optimization Heuristics, Resilience in Production Systems.

1 Extended Abstract

The transformation of production systems under Industry 4.0 and the emerging paradigm of Industry 5.0 has introduced unprecedented complexity into shop-floor management. One of the most challenging scenarios arises in flow shop environments with missing operations, where jobs may skip certain stages depending on customer specifications. This variability reflects the essence of mass customization, as production routes

are no longer uniform and decision-makers must adapt schedules to heterogeneous demands [1]. Addressing this problem requires balancing multiple objectives simultaneously: minimizing makespan to optimize resource utilization, reducing total tardiness to meet customer deadlines, and lowering total completion time to minimize work-in-progress. Recent research has demonstrated that multiobjective evolutionary algorithms (MOEAs) (including NSGA-II, NSGA-III, MOEA/D, SPEA2, and AGEMOEA-II) are competitive approaches for tackling such NP-hard problems [2]. Computational experiments confirm that these algorithms can efficiently approximate Pareto-optimal schedules across diverse instances, while also revealing the significant impact of missing operations on performance metrics. From a managerial perspective, this implies that production systems must be designed with agility and resilience, capable of dynamically reconfiguring sequences and adapting to variability without sacrificing efficiency [3].

Complementing this systemic perspective, another stream of research focuses on additive manufacturing (AM), a technology that typifies customization by enabling the creation of highly complex and individualized geometries. Unlike subtractive methods, AM builds objects layer by layer, allowing unique designs without prohibitive costs. Yet, this flexibility introduces new planning challenges: machines can process multiple parts simultaneously, but grouping parts into jobs requires nesting logic that accounts for geometric constraints such as volume, area, and height [4]. The integrated nesting and scheduling problem is NP-hard, demanding hybrid approaches that combine heuristics with mathematical programming [5]. Studies show that heterogeneous builds, where jobs differ in average volumes or heights, not only improve makespan efficiency by approximately 2% but also drastically reduce computational time, from more than 1,100 seconds to just 22 seconds in large instances. This finding provides practitioners with actionable rules for grouping parts, ensuring that AM systems can scale to industrial levels while remaining responsive to customer-specific demands [6].

In this presentation, both approaches will be analyzed jointly: the flow shop with missing operations and the additive manufacturing nesting/scheduling problem. This dual perspective allows us to compare how different production technologies and optimization models confront the challenges of personalization. By examining them side by side, we can better understand the managerial implications of variability (whether expressed as skipped operations in flow shops or heterogeneous builds in AM) and derive practical insights for improving efficiency, resilience, and competitiveness in customized production systems.

Taken together, these two research streams illuminate complementary facets of personalized production management. The flow shop study highlights the systemic and multiobjective dimension, where variability in operation sequences requires evolutionary approaches capable of balancing efficiency, customer service, and throughput. The AM study emphasizes the geometric and batching dimension, showing how heuristic nesting strategies directly influence scheduling quality and computational feasibility. Both underscore the necessity of hybrid optimization frameworks that integrate heuristics, mathematical programming, and metaheuristics to cope with NP-hard problems in real industrial contexts.

From a broader managerial standpoint, the convergence of these works suggests several implications. First, heterogeneity can be leveraged as a resource: whether in the form of missing operations in flow shops or heterogeneous builds in AM, variability tightens optimization models and improves computational performance. Second, resilience and agility become central performance indicators: production systems must not only minimize makespan but also ensure robustness against disruptions, urgent orders, or unexpected failures. Third, practical rules and policies are as valuable as sophisticated algorithms: by translating complex optimization insights into simple heuristics (e.g., grouping parts by decreasing volume or balancing job heterogeneity), decision-makers can implement strategies that are both effective and operationally feasible.

Ultimately, these contributions enrich the discourse on Industry 4.0/5.0 by demonstrating that mass customization requires new forms of production planning knowledge. Traditional scheduling models, designed for standardized environments, are insufficient when confronted with the variability inherent in personalized production. Instead, the future of operations management lies in hybrid, data-driven, and adaptive approaches that combine theoretical rigor with practical applicability. By bridging the contexts of flow shop scheduling with missing operations and additive manufacturing nesting problems, these works provide a unified vision of how optimization can support the transition from mass production to mass customization, ensuring that industrial systems remain competitive, resilient, and customer-oriented in the era of smart manufacturing.

References

1. Rossit, D. A., Toncovich, A. A., Rossit, D. G., & Nesmachnow, S. (2020). Solving a flow shop scheduling problem with missing operations in an Industry 4.0 production environment. Journal of Project Management, 6, 33-44.
2. Ojstersek, R., Brezocnik, M., & Buchmeister, B. (2020). Multi-objective optimization of production scheduling with evolutionary computation: A review. *International Journal of Industrial Engineering Computations*, *11*(3), 359-376.
3. Rossit, D., Rossit, D., & Nesmachnow, S. (2024). Enhancing mass customization manufacturing: Multiobjective metaheuristic algorithms for flow shop production in smart industry. *SN Computer Science*, *5*(6), 782.
4. Kucukkoc, I. (2019). MILP models to minimise makespan in additive manufacturing machine scheduling problems. *Computers & Operations Research*, *105*, 58-67.
5. Oh, Y., Witherell, P., Lu, Y., & Sprock, T. (2020). Nesting and scheduling problems for additive manufacturing: A taxonomy and review. *Additive Manufacturing*, *36*, 101492.
6. Rodriguez, J., & Rossit, D. (2025). Nesting and Scheduling in Additive Manufacturing: The Impact of Practical Nesting Strategies on Overall Makespan Efficiency. *IET Collaborative Intelligent Manufacturing*, *7*(1), e70036.

Modelo de estimación de tiempos de llegada en transporte público urbano de Cuernavaca mediante redes neuronales artificiales

Edelmira Tapia[1], Pedro Moreno[1 [0000−0002−2811−5331]],
Carlos Torres[2 [0000−0001−6187−4519]],
Carmen Peralta[1 [0009−0001−8674−8209]], and
Jose Hernández[1 [0000−0002−5184−0005]]

[1] Universidad Autónoma del Estado de Morelos, Cuernavaca, 62209, México
{pmoreno, carmen.peralta, jose_hernandez}@uaem.mx
[2] Universidad Juárez Autónoma de Tabasco, Cunduacán, 86690, México
enrique.torres@ujat.mx

Abstract. El transporte público urbano es un componente esencial para la movilidad sostenible en las ciudades. Sin embargo, uno de los principales problemas que enfrentan los usuarios es la falta de información precisa sobre los tiempos de llegada en cada parada, lo que genera incertidumbre y disminuye la calidad del servicio. Este trabajo presenta un enfoque basado en aprendizaje supervisado mediante redes neuronales artificiales para estimar el tiempo de viaje entre dos paradas de autobús en la ciudad de Cuernavaca, México. El estudio utiliza un conjunto de datos compuesto por 2 073 registros de viaje, considerando variables como hora de salida y llegada, velocidad, distancia, número de semáforos y topes, condiciones climáticas y eventos inesperados. Los resultados muestran que el modelo propuesto alcanza un coeficiente de determinación (R^2) superior a 0.9995, superando significativamente el método base de regresión lineal. Este enfoque demuestra el potencial de las redes neuronales artificiales como herramienta para optimizar la planificación del transporte urbano y mejorar la experiencia del usuario.

Keywords: estimación de tiempos de llegada · transporte público · redes neuronales artificiales · movilidad urbana.

1 Introducción

La movilidad urbana eficiente es un pilar fundamental para el desarrollo sostenible de las ciudades modernas [1]. El transporte público contribuye a reducir la congestión vehicular, las emisiones de gases de efecto invernadero y la dependencia del automóvil particular [6]. Sin embargo, en ciudades como Cuernavaca, la falta de información confiable sobre los tiempos de llegada de los autobuses genera inconvenientes para los usuarios, quienes enfrentan largos periodos de espera y dificultades para planificar sus desplazamientos [5].

La estimación precisa de los tiempos de llegada es un desafío complejo debido a factores dinámicos como el tráfico, las condiciones meteorológicas, la variabilidad en el abordaje de pasajeros y eventos inesperados [4]. En este contexto, los sistemas de transporte inteligente (ITS) y las técnicas de inteligencia artificial ofrecen soluciones prometedoras para mejorar la confiabilidad del servicio [7]. Este artículo propone un modelo basado en redes neuronales artificiales (ANN, por sus siglas en inglés) para predecir el tiempo de viaje entre paradas, contribuyendo a la optimización del transporte público en Cuernavaca. Asimismo, el trabajo propuesto contribuye al desarrollo de sistemas de transporte inteligente, promoviendo la movilidad sostenible y la calidad del servicio en la ciudad de Cuernavaca.

2 Metodología

El enfoque propuesto utiliza un modelo de ANN tipo perceptrón multicapa entrenado mediante aprendizaje supervisado [3]. El objetivo es predecir el tiempo total de viaje entre dos paradas consecutivas, considerando un conjunto de variables que influyen en la duración del trayecto.

Conjunto de datos. Los datos fueron recolectados en la ruta 13 del transporte público de Cuernavaca, que conecta la Universidad Autónoma del Estado de Morelos con Walmart Jiutepec. El periodo de observación abarcó del 1 de octubre de 2024 al 20 de febrero de 2025, registrando 26 recorridos completos y generando un total de 2 073 registros para entrenamiento y validación [5]. Las variables incluidas en el modelo son: hora de salida y llegada, distancia, tiempo de viaje, velocidad, número de topes y semáforos, tipo de día, turno, clima y eventos inesperados.

Preparación y normalización. Se aplicaron procesos de limpieza para eliminar registros erróneos y normalización para escalar las variables [2]. Se construyó una matriz de correlación para identificar relaciones significativas, destacando la correlación positiva entre distancia y tiempo de viaje, y la negativa entre velocidad y duración del trayecto.

Arquitectura del modelo. El modelo ANN consta de los siguientes componentes: *i*) capa de entrada: 12 neuronas (una por variable), *ii*) capas ocultas: 3 a 4 capas con funciones de activación ReLU, y *iii*) capa de salida: una neurona para estimar el tiempo total. La optimización se realizó mediante el algoritmo Adam y ajuste de hiperparámetros con GridSearch.

3 Resultados y discusión

El modelo se entrenó con una división 80-20 entre datos de entrenamiento y prueba, aplicando validación cruzada. Las métricas utilizadas fueron MAE, RMSE y R^2. Los resultados más relevantes son: MAE: 0.5951 segundos, RMSE:

0.7932 segundos y R^2: 0.9998. Estos valores indican una alta precisión en la estimación de tiempos de llegada, con un error promedio inferior al 1%. Comparado con el método base de regresión lineal multivariante, el modelo ANN mejora el rendimiento en un 14.99%. Las predicciones de la ANN comparadas con los valores reales muestran una alineación casi perfecta, evidenciando la capacidad del modelo para generalizar sin sobreajuste.

La implementación de la ANN para estimar tiempos de llegada en transporte público urbano demuestra ser una solución eficaz frente a métodos tradicionales. La alta precisión obtenida se atribuye a la capacidad de las redes neuronales para capturar relaciones no lineales entre variables y adaptarse a condiciones dinámicas. Este enfoque puede integrarse en sistemas ITS para ofrecer información en tiempo real a los usuarios, reduciendo la incertidumbre y mejorando la experiencia de viaje. Sin embargo, el modelo depende de la calidad y representatividad de los datos. Factores como la variabilidad estacional, eventos extremos y cambios en la infraestructura pueden afectar su desempeño.

4 Conclusiones y trabajo futuro

El estudio confirma que las redes neuronales artificiales son una herramienta poderosa para la estimación de tiempos de llegada en transporte público urbano. El modelo propuesto alcanzó un R^2 superior a 0.9995, superando significativamente el método base y demostrando su aplicabilidad en entornos reales. El trabajo futuro plantea integrar datos en tiempo real mediante GPS y dispositivos de Internet de las Cosas. Asimismo, se plantea incorporar variables adicionales como número de pasajeros y condiciones de tráfico, evaluar arquitecturas más complejas para capturar dependencias temporales, y generar datasets sintéticos para mejorar la robustez del modelo.

References

1. Al Suleiman, S., Cortez, A., Monzón, A., Lara, A.: How to improve public transport usage in a medium-sized city: key factors for a successful bus system. Eur. Transp. Res. Rev. **15**(47) (2023). https://doi.org/10.1186/s12544-023-00616-y
2. Barbierato, E., Gatti, A.: The challenges of machine learning: A critical review. Electronics **13**(2) (2024). https://doi.org/10.3390/electronics13020416
3. Bishop, C.M.: Pattern Recognition and Machine Learning. Springer (2006)
4. Lam, C.T., Ng, B., Leong, S.H.: Prediction of bus arrival time using real-time online bus locations. In: International Conference on Communication Technology. pp. 473–478 (2019). https://doi.org/10.1109/ICCT46805.2019.8947251
5. León Sosa, S.E., Hernández Báez, I., Morales Morales, C., Maldonado Martínez, H.J.: El traslado de pasajeros en transporte público en el área metropolitana de cuernavaca morelos. ANFEI Digital **6**(11), 1–11 (2019)
6. Nesmachnow, S., Hipogrosso, S.: Transit oriented development analysis of parque rodó neighborhood, montevideo, uruguay. World Development Sustainability **1**, 100017 (2022). https://doi.org/10.1016/j.wds.2022.100017
7. Rodrigue, J.: The Geography of Transport Systems. Routledge (2020). https://doi.org/10.4324/9780429346323

A temporal perspective on optimization for strategic, tactical, and operational decisions in smart cities

Diego Rossit[1][0000-0002-8531-445X] and Sergio Nesmachnow[2][0000-0002-8146-4012]

[1] Department of Engineering, INMABB, Universidad Nacional del Sur-CONICET, Argentina
diego.rossit@uns.edu.ar
[2] Universidad de la República, Uruguay
sergion@fing.edu

Abstract. Smart cities rely on data-driven decision-making to improve the performance and sustainability of urban services. This article examines how optimization techniques support urban planning across three complementary horizons: strategic decisions, such as locating waste accumulation infrastructure; tactical decisions, including the synchronization of public transport timetables; and operational decisions, exemplified by scheduling residential energy loads. By analyzing these domains, optimization not only enhances the efficiency of individual services but also ensures coherence among long-term investments, resource coordination, and real-time user interactions. This temporal perspective highlights the role of optimization as a unifying mechanism for designing resilient, efficient, and citizen-centered smart urban systems.

Keywords: Smart cities; Optimization; Strategic Planning; Tactical Decision-making; Operational Decisions

1 Decision-making in smart cities

Urban environments face unprecedented complexity driven by population growth, environmental pressures, digital transformation, and rising expectations regarding service quality and sustainability. A fundamental, yet often overlooked, characteristic of smart city planning is that decisions are not homogeneous over time. Instead, they operate along three interdependent planning horizons—strategic, tactical, and operational—each associated with distinct objectives, decision-makers, data availability, and model requirements. Understanding these temporal layers is essential to deploying optimization techniques appropriately and avoiding solutions that are technically correct but contextually invalid.

The strategic horizon encompasses long-term planning decisions whose implications persist over years or decades. They involve infrastructure, regulatory frameworks, and technology adoption choices that require significant investment and are difficult to reverse. These decisions typically employ mathematical or multi-objective models capable of balancing economic, environmental, and social criteria. The tactical horizon transforms strategic intentions into medium-term operational policies. It deals with resource coordination, scheduling, and regulatory rules that maintain service functional-

ity under existing structural constraints. Tactical decisions are more flexible than strategic ones and can be revisited periodically. The operational horizon addresses decisions executed on short time scales—from daily routines to real-time adjustments. These decisions interact closely with user behavior and environmental variability, and increasingly rely on data-driven and adaptive optimization methods. The following sections illustrate each horizon through three emblematic smart city domains where optimization is not only applicable, but decisive.

2 Illustrative examples of smart cities decisions for different planning horizons

One of the most representative strategic decisions in smart cities is defining the location and sizing of waste accumulation points. This decision fundamentally shapes the solid waste management system for years, since infrastructure placement affects service accessibility, environmental impact, and subsequent routing operations. Actors involved include municipal planners, environmental agencies, waste collection operators, and local communities, each with conflicting interests: residents demand proximity and fair access, operators seek efficiency and reduced travel distances, environmental departments aim to prevent overflow, noise, and pest proliferation, and budget authorities limit capital investment. Constraints typically include: zoning restrictions, walking distance thresholds, acceptable landfill capacity, container availability, neighborhood complaint thresholds, and long-term demographic projections. The decision is strategic because modifying infrastructure after deployment entails high sunk costs, legal procedures, and public resistance. Optimization contributes by exploring trade-offs between conflicting goals such as: minimizing the number of bins versus maximizing equitable access, reducing environmental exposure versus lowering collection costs, and limiting container volume versus allowing demand fluctuations. Real-world studies have shown that optimized layouts can reduce walking distances by more than 30%, avoid infrastructure saturation, and improve equity without increasing costs [1][4]. Moreover, strategic decisions at this level profoundly constrain tactical routing decisions, demonstrating the temporal dependency inherent to smart city systems.

Once infrastructures and service areas are defined, cities face tactical decisions that determine how resources are used within those constraints. A paradigmatic example in urban mobility is deciding whether two bus lines should be synchronized to guarantee passenger transfers below a maximum waiting time threshold. Unlike strategic placement of assets, synchronization does not alter the transport network, but adjusts how it is operated. This decision requires analyzing: passenger flow patterns, peak and off-peak demand periods, available rolling stock, timetable feasibility, drivers' labor constraints, the elasticity of user satisfaction regarding waiting times. Authorities must evaluate whether coordination yields net social benefit, considering that improved connectivity could: reduce total travel time, increase ridership, improve network attractiveness, require additional buses or longer cycles. From an optimization perspective, the problem is inherently multi-objective: maximizing synchronization events while minimizing operational costs and respecting resource limits. Heuristic and exact methods

have demonstrated that optimized schedules can improve synchronization rates by more than 60% and reduce transfer waiting times by up to 57%, outperforming administrative practices [2]. This example captures the essence of tactical decisions: they refine the system without reshaping it, and their success depends on coherently interpreting strategic objectives—such as promoting public transport usage—into feasible operational protocols.

At the operational level, decisions unfold in short time horizons and involve direct interaction with users. In residential energy planning, a representative operational decision is determining when to activate deferrable appliances—such as washing machines, dishwashers, or thermal conditioning systems—to minimize electricity costs while maintaining user comfort. This decision is influenced by: hourly tariff variations, power limits contracted with the utility, thermal inertia of devices, user preferences (e.g., "the laundry must be ready before 7 AM"), and the statistical variability of consumption patterns. Unlike strategic and tactical decisions, operational choices incorporate uncertainty from human behavior and real-time sensor data. Thus, they require adaptive optimization, often based on stochastic modeling, simulation-optimization techniques, or evolutionary algorithms. Empirical studies show that optimized schedules can reduce energy bills without altering user routines, and when combined with dynamic pricing, shift a significant portion of demand to off-peak hours, increasing grid stability and reducing carbon emissions [3]. Despite their minimal scope compared to infrastructure placement, operational decisions collectively exert substantial impact on sustainability goals.

3 Final comments

The three examples—waste infrastructure (strategic), bus timetable synchronization (tactical), and energy scheduling (operational)—confirm that optimization serves as a temporal integrator in smart cities. Strategic choices define capabilities; tactical rules exploit them; operational actions execute and refine them. Designing intelligent urban systems requires embedding optimization not merely in isolated tools, but across the entire temporal decision cycle.

References

1. Rossit, D., Toutouh, J., Nesmachnow, S.: Exact and heuristic approaches for multi-objective garbage accumulation points location in real scenarios. Waste Management 105, 467–481 (2020)
2. Risso, C., Nesmachnow, S., Rossit, D.: Smart mobility for public transportation systems: Improved bus timetabling for synchronizing transfers. Communications in Computer and Information Science 1706, 158–172 (2022)
3. Nesmachnow, S., Rossit, D., Toutouh, J., Luna, F.: An explicit evolutionary approach for multiobjective energy consumption planning considering user preferences in smart homes. International Journal of Industrial Engineering Computations 12(4), 365–380 (2021)
4. Rossit, D., Nesmachnow, S.: Waste bins location problem: A review of recent advances. Journal of Cleaner Production 342, 130793 (2022)

Modelado y evaluación de soluciones pasivas en edificaciones para ciudades sostenibles

Carlos Torres-Aguilar *[1][0000−0001−6187−4519], Pedro Moreno[2][0000−0002−2811−5331], Sergio Nesmachnow[3][0000−0002−8146−4012], Karla María Aguilar-Castro[1][0000−0003−2611−2820], and Edgar Vicente Macias-Melo[1][0000−0003−0107−766X]

[1] Universidad Juárez Autónoma de Tabasco, México
enrique.torres@ujat.mx, karla.aguilar@ujat.mx, edgar.macias@ujat.mx
[2] Universidad Autónoma del Estado de Morelos, México
pmoreno@uaem.mx
[3] Universidad de la República, Uruguay
sergion@fing.edu.uy

Abstract. En este artículo se presenta una metodología de evaluación y comparación de modelos basados en balances globales de energía y en dinámica de fluidos computacional aplicada a chimeneas solares y paneles fotovoltaicos. Se compararon el tiempo de cómputo y los niveles de información obtenidos mediante las diferentes metodologías. Aunque la diferencia en el tiempo de cómputo es notablemente mayor en los enfoques de dinámica de fluidos computacional, el nivel de detalle de la solución permite realizar análisis más extensos y la detección de puntos críticos. Se concluye la necesidad de combinar ambos enfoques, dando como resultado metodologías híbridas para la modelación de los fenómenos de transferencia de calor y de mecánica de fluidos involucrados en los presentes sistemas.

Keywords: Sistema pasivo · tiempo de cómputo · balance global de energía · dinámica de fluidos computacional.

1 Introducción

En los últimos años, la demanda energética ha aumentado un 20% desde el 2010 de acuerdo con la Agencia International de Energía [2], además, el uso de combustibles fósiles para la generación eléctrica representa cuatro quintos del total de las fuentes de energía para satisfacer la demanda energética total. Los sectores industrial y residencial se posicionan como los que presentan la mayor demanda energética; tan solo en el sector residencial se emplea un tercio de la energía global [1]. Por lo tanto, como alternativas de solución se han desarrollado estrategias como la implementación de sistemas pasivos para ventilación y confort térmico, así como generación de energía para el funcionamiento de ciudades sostenibles.

* Correspondence: enrique.torres@ujat.mx

Entre las alternativas presentes se encuentra la implementación de chimeneas solares (ChSo) para inducir la ventilación natural en las edificaciones, lo cual también favorece el confort de los ocupantes [3]. En cuanto a la generación de energía por paneles fotovoltaicos (PV), su estudio se ha centrado en optimizar parámetros para aprovechar al máximo la energía recibida, sin depender de aditamentos mecánicos que incrementen la radiación incidente. Sin embargo, la implementación física de tales sistemas requiere tiempo y recursos considerables. Por esta razón, ambos sistemas deben ser estudiados desde la perspectiva de la modelación mediante balances globales de energía (BEG) y modelos diferenciales, utilizando técnicas de dinámica de fluidos computacional (DFC).

A continuación, se expone una metodología de evaluación y comparación de los modelos BEG y DFC de una ChSo y de un PV en condiciones reales. Se describen los resultados de la comparación considerando las limitaciones de cada enfoque y la información proporcionada .

2 Metodología

2.1 Descripción del problema

Para comparar ambos enfoques, se estableció que ambos sistemas (ChSo y PV) se sometieran a condiciones climáticas reales. Por lo que, desde la formulación del modelo BEG y DFC se consideró un enfoque transitorio. Ambos sistemas se evaluaron en condiciones climáticas de Villahermosa, Tabasco. Los datos climáticos de ambos modelos se obtuvieron de las estaciones meteorológicas de la Comisión Nacional del Agua. Se eligieron las condiciones climáticas del día más cálido y frío de cada mes del año 2024 para evaluar ambos enfoques.

2.2 Modelos BEG y DFC

La Ec. 1 presenta de manera general la estructura del balance de energía; esto implica que la información obtenida de este tipo de modelos se limita a las temperaturas y flujos de calor obtenidos directamente de la solución del sistema de ecuaciones resultante. La Ec. 2 presenta la forma general de la ecuación de convección-difusión empleada para representar los efectos de la conservación de masa, cantidad de movimiento y energía; por lo que, además de determinar las isotermas, es posible determinar el campo de presiones y campo vectorial de velocidades en diferentes geometrías.

$$\rho\, c_p \frac{\partial T}{\partial t} \;=\; \sum q_{inputs} \;-\; \sum q_{outputs} \tag{1}$$

$$\frac{\partial(\rho\phi)}{\partial t} + \frac{\partial(\rho u_j \phi)}{\partial x_j} = \frac{\partial}{\partial x_j}\left[\Gamma \frac{\partial \phi}{\partial x_j}\right] + S \tag{2}$$

Para más detalles sobre los modelos descritos en las Ecs. 1 y 2 consultar el estudio de Torres-Aguilar et al. [4]. Los modelos de la ChSo y el PV consideran

el acoplamiento de los fenómenos de conducción, convección y radiación de calor, materiales con propiedades no lineales como materiales de cambio de fase (PCM) y son capaces de considerar, además de la temperatura y la radiación solar, los efectos de la velocidad del viento, la humedad relativa y la presión atmosférica. La mayoría de las variables meteorológicas mencionadas suele no considerarse en la literatura para los modelos actuales de la ChSo y de los sistemas PV, además del uso de un enfoque multinodal en materiales masivos para modelos transitorios. En cuanto al modelo radiativo, se consideró un enfoque de intercambio radiativo superficial, por lo que se evalúa incluir a futuro un modelo de radiación en medio participante.

Los códigos de los modelos BEG y DFC se desarrollaron en ANSI C99 y se compilaron con GCC 7.5.0 en GNU/Linux (Ubuntu 18.05, 64 bits). Los resultados se analizaron mediante una herramienta desarrollada en Python 3.11 para su tratamiento y visualización.

En ambos enfoques se realizó un estudio de independencia de malla espacial y temporal para determinar el paso de tiempo adecuado y el número de divisiones necesarias para los elementos de mayor masividad térmica del modelo BEG.

3 Validación experimental y resultados

Los resultados del comportamiento de la ChSo y el PV fueron validados con datos reportados en literatura, obteniéndose valores menores al 10% de diferencia porcentual entre los datos teóricos y los experimentales [5].

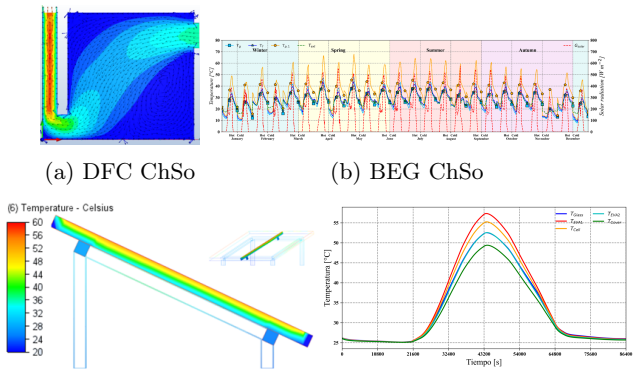
En la Tabla 1 se presentan los tiempos de cómputo obtenidos con los distintos enfoques para completar un día de evaluación del sistema. Aunque la diferencia de tiempo es notable. Los resultados obtenidos revelan que aunque la implementación en menor tiempo de un enfoque BEG provee resultados factibles para la toma de decisiones, la diferencia porcentual con respecto a los resultados experimentales puede incrementarse hasta un 10% adicional comparada con el estudio con un enfoque DFC.

Table 1: Comparación tiempo de cómputo para un día de simulación

Modelo físico	**Enfoque DFC Tiempo (horas)**	**Enfoque BEG Tiempo (horas)**
Chimenea solar	20	0.020
Panel fotovoltaico	72	0.001

Una de las principales desventajas presentadas desde una solución con enfoque DFC fue el notable tiempo de cómputo requerido para obtener un solo día de comportamiento del fenómeno real. Como se observa en la Fig. 1 aunque el enfoque BEG carece de mayor detalle en la solución numérica y resultados obtenidos, la información es suficiente para el análisis requirido.

Los modelos actuales permiten evaluar el impacto en la ventilación natural y el confort térmico en las edificaciones al utilizar sistemas de aprovechamiento de



(a) DFC ChSo (b) BEG ChSo

(c) DFC PV (d) BEG PV

Fig. 1: Temperaturas de ChSo y PV en diferentes enfoques

energías renovables y generación de energía. El modelo actual de la ChSo permite evaluar su impacto en la atención de necesidades industriales como mejora de procesos de soldadura y fundición donde se requieren niveles adecuados de renovación de aire, al igual que en zonas no residenciales como oficinas u escuelas que son susceptibles a síndromes como el del "edificio enfermo" debido a una incorrecta ventilación. En cuanto a los PV, el modelo actual permite realizar parametrizaciones para identificar las mejores configuraciones en zonas residenciales y no residenciales que maximicen la ganancia de energía, debido a su orientación y a las variaciones constantes de las condiciones climáticas.

Los resultados muestran la necesidad de evaluar una nuevo enfoque de solución que permita mezclar ambas metodologías como un esquema híbrido. Esto permitiría aplicar más detalles a partir de modelos diferenciales en los puntos críticos y obtener un balance general de un área de mayor dimensión, reduciendo el tiempo de cómputo sin comprometer los resultados numéricos respecto a la realidad de los fenómenos involucrados.

4 Trabajo a futuro

Como futura ruta de trabajo, se evaluará el efecto de la metodología híbrida sobre la precisión de los resultados frente a su desempeño computacional. En primer lugar, se deben evaluar los fenómenos con mayor no linealidad; posteriormente, elegir el método de discretización de la ecuación gobernante del fenómeno y, finalmente, implementar y validar experimentalmente los sistemas de interés a analizar. La elección del método de discretización obedece a qué secciones del sistema se estarán evaluando desde un enfoque con dependencia espacial y temporal (CFD), con lo cual la zonas de menor impacto se considerarán bajo enfoque BEG e incorporando correlaciones obtenidas experimentalmente que representen efectos tales como la razón de transferencia de energía entre sólidos y fluidos (coeficientes convectivos) y el regimen de flujo (laminar o turbulento).

References

1. Agency, I.E.: Tracking clean energy progress 2023 (2023), https://www.iea.org/reports/tracking-clean-energy-progress-2023, licence: CC BY 4.0
2. Agency, I.E.: World energy outlook 2025 (2025), https://www.iea.org/reports/world-energy-outlook-2025, licence: CC BY 4.0 (report); CC BY NC SA 4.0 (Annex A)
3. Serageldin, A.A., Abdelrahman, A.K., Ookawara, S.: Parametric study and optimization of a solar chimney passive ventilation system coupled with an earth-to-air heat exchanger. Sustainable Energy Technologies and Assessments **30**, 263–278 (2018)
4. Torres-Aguilar, C., Moreno, P., Rossit, D., Nesmachnow, S., Aguilar-Castro, K.M., Macias-Melo, E.V., Hernández-Callejo, L.: Forecasting performance indicators of a single-channel solar chimney using artificial neural networks. Computer Modeling in Engineering & Sciences pp. 1–23 (2025)
5. Torres-Aguilar, C., Arce, J., Xamán, J., Macias-Melo, E.: Experimental study and numerical analysis of radiative losses of single-channel solar chimney. Journal of Building Physics **46**(3), 340–371 (2022)

UNIVERSIDAD
DE LA REPÚBLICA
URUGUAY

CES
CENTRO DE
ENSAYOS
DE SOFTWARE