LLMs FOR ECOLOGY

Introduction to Large Language Models

With PRACTICAL SESSION

Systematic Literature Review in Ecology

Today's Menu

01

The Parrot

What is an LLM · How it works

02

The Controls

Tokens · Parameters · Models

03

The Kitchen

Local vs API · Real use case

04

The Practice

Screening · Extraction · Validation

SECTION 01

A Stochastic Parrot?

Bender et al., 2021 — What an LLM really is

What is a Large Language Model?

A neural network trained on massive text to **predict the next token** .
Everything else (summarizing, classifying, answering) emerges from this single task.

What emerges

- Answering questions
- Summarizing text
- Classifying documents
- Extracting structured data (JSON)

What it does NOT do

- "Know" facts — reproduces patterns
- Verify its own outputs
- Reason like a scientist
- Access data beyond training

The Stochastic Parrot

An LLM stitches together words based on probabilistic patterns — without understanding their meaning.

- It has seen billions of sentences during training
- It learned which words tend to follow which
- It does NOT have a model of the world
- It cannot tell if its output is true or false

"Not a knowledge base — a language probability machine."

Key Implications for Ecologists

It can answer brilliantly about great tit phenology and hallucinate references about a rare tropical species in the same conversation.

Language bias

Tokenizers optimized on English.
French = 30–50% more tokens.

Taxonomic bias

Charismatic megafauna OK.
Rare tropical species = gaps.

Geographic bias

North America & Europe
dominate training data.

SECTION 02

The Controls

Tokens, parameters, and choosing a model

Tokens — What the Model Sees

The model reads integers (tokens), not text. BPE tokenizer splits text into subword units.

1 token

≈ 0.75 word

10K tokens

≈ 15-page article

128K–1M

context window range

Generation Parameters

temperature

Controls randomness.
0 = deterministic (use for classification)
1 = creative (brainstorming)

top_p

Nucleus sampling (0–1).
Keeps tokens up to cumulative prob p.
Default: 0.95

max_new_tokens

Maximum output length.
Too low = truncated invalid JSON!

seed

Reproducible generation.
Only same model + version + hardware.

```
TODAY:  temperature=0  |  top_p=1  |  seed=42
```

The LLM Explosion

Claude **Gemini** **Mistral** **DeepSeek** **Llama**

	GPT-4o	GPT-5	Local
Context	128K	400K	8K–32K
Input $/1M	$2.50	$1.25	**Free**
Output $/1M	$10.00	$10.00	**Free**
Reproducibility	Low	Low	**Excellent**

SECTION 03

The Kitchen

Local vs API, prompting, and a real use case

Local Model vs. Cloud API

Cloud API

- Best models (GPT-5, Claude...)
- Pay per token
- Data sent to third party
- Models retired without notice
- Carbon footprint opaque

Local Model

- Full data control — nothing leaves
- Excellent reproducibility
- Free after setup
- Needs GPU ≥ 8 GB VRAM
- Lower perf on complex tasks

For research: local = reproducibility + privacy + free. API = risk of model retirement.

The Prompt Matters

"The better you talk to the model, the better it responds."

Vague

```
"Is this abstract relevant
to my review?"
```

Model has no criteria.

Structured

```
Research question
5 inclusion criteria
Borderline instruction
Output format (1 word)
```

Model knows exactly what to check.

A Real Use Case

Invasive species impact — extracting structured data from scientific papers

Chain of prompts

Species ID
Geography
Ecosystem
Impact type

→

API processing

PDFs ⟶ GPT-5
via API scripts
+ multithreading

→

Structured table

40 columns
10 categories
Machine-readable

7,065	10h	$800	66%	30
articles	total time	API cost	Jaccard	failures

Source: Prospective IA & Ecologie, CNRS 2025

Hallucinations — The Central Risk

The model generates text that is fluent, plausible, and completely wrong.

Invented references

Plausible author + journal + year — doesn't exist.

Wrong numbers

30 years becomes 31. Sites confused with individuals.

Taxonomy errors

Correct genus, wrong species. Plausible to non-specialists.

Always manually validate a sample before trusting at scale.

SECTION 04

The Practice

Hands-on: screening, extraction, validation

Today's Scenario

"Effects of climate change on the breeding phenology of birds in Europe"

Inclusion criteria — all 5 must be met:

1. Empirical study (not reviews, not theoretical)
2. About birds (class Aves)
3. In Europe (EU + UK + Norway + Switzerland)
4. Breeding phenological trait (laying date, arrival...)
5. Explicit link with a climatic variable

30 abstracts with traps: wrong taxon, wrong region, no phenology, reviews disguised as empirical...

Part 1 — Abstract Screening

What you'll do

- Write your own screening prompt
- Run it on 30 abstracts
- Get INCLUDE / EXCLUDE for each
- Compute confusion matrix vs gold standard
- Identify which trap types fooled the LLM
- Discuss: how to improve the prompt?

Trap types in the corpus

- Wrong taxon, identical syntax
- Correct birds, outside Europe
- Birds + Europe but no phenology
- Birds + phenology but no climate
- Not empirical (reviews, R packages)
- Keyword bait (right words, wrong topic)
- Genuinely ambiguous cases

Part 2 — Metadata Extraction

JSON schema to extract

```
taxons
country_or_region
study_type
duration_years
sample_size
phenological_trait
climatic_variable
```

Key rules

- **If info not in abstract → null**
- **Never fabricate data**
- Parse JSON safely (markdown wrappers, trailing commas...)
- Result = pandas DataFrame for filtering

Part 3: you'll compare LLM extractions side-by-side with original abstracts to catch hallucinations.

What's Next — RAG

Today = abstracts only. For full-text analysis: Retrieval-Augmented Generation.

1 Index

Split documents into chunks.
Compute embeddings.
Store in vector DB.

2 Retrieve

For each query, find the
k most similar chunks.

3 Generate

Feed LLM question +
chunks as context.
Citable answers.

`LangChain | LlamaIndex | ChromaDB / FAISS | sentence-transformers | Ollama`

LET'S GO

Open the Notebook

`temperature=0 | top_p=1 | seed=42`

Reproducibility, not creativity.