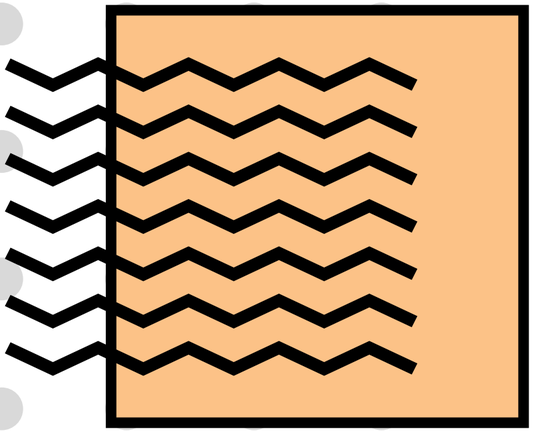
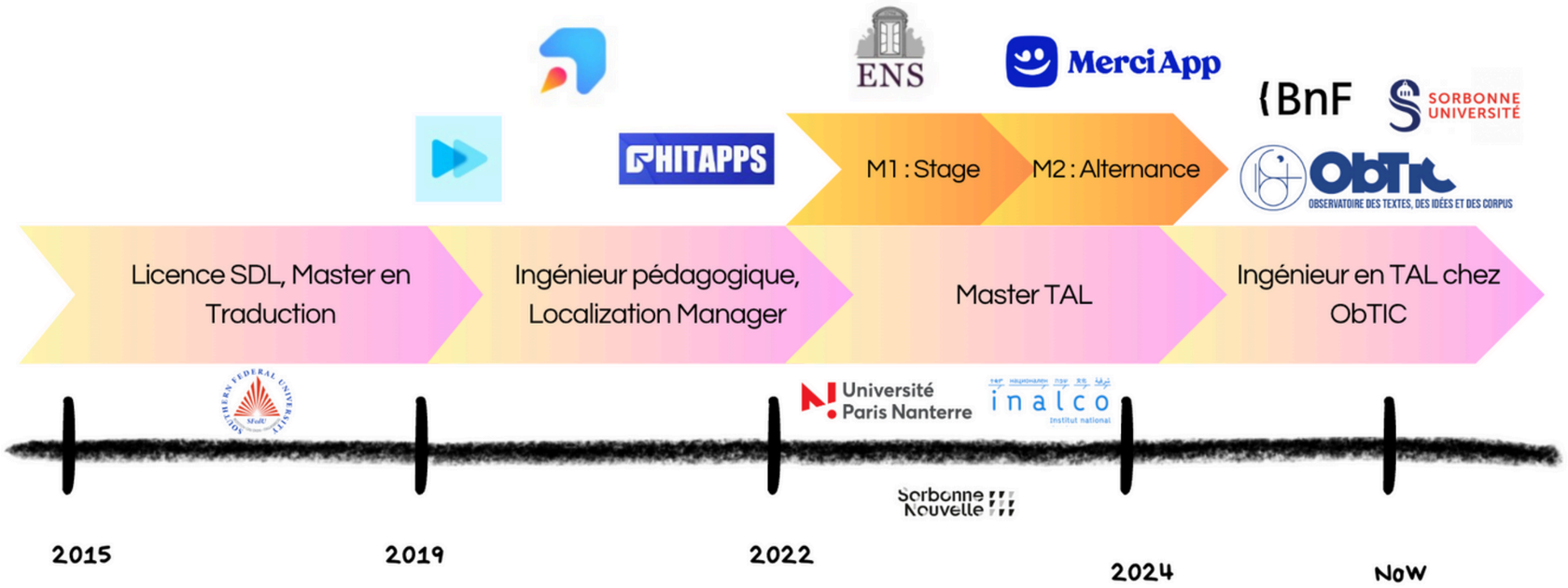


TRAITEMENT DU LANGAGE NATUREL AVEC SPACY

Préparé par : Mikhail Biriuchhinskii
Ingénieur en Traitement automatique des langues
ObTIC

MON PARCOURS



Mikhail Biriuchinskii (He/Him)

NLP Engineer | Data Scientist

Paris, Île-de-France, France · **Contact info**

500+ connections

Au cas où, voici ma page Linkedin.

PLAN DE L'ATELIER

01

THÉORIE

Les notions de base

02

PRATIQUE

Google Colab

03

CONCLUSION

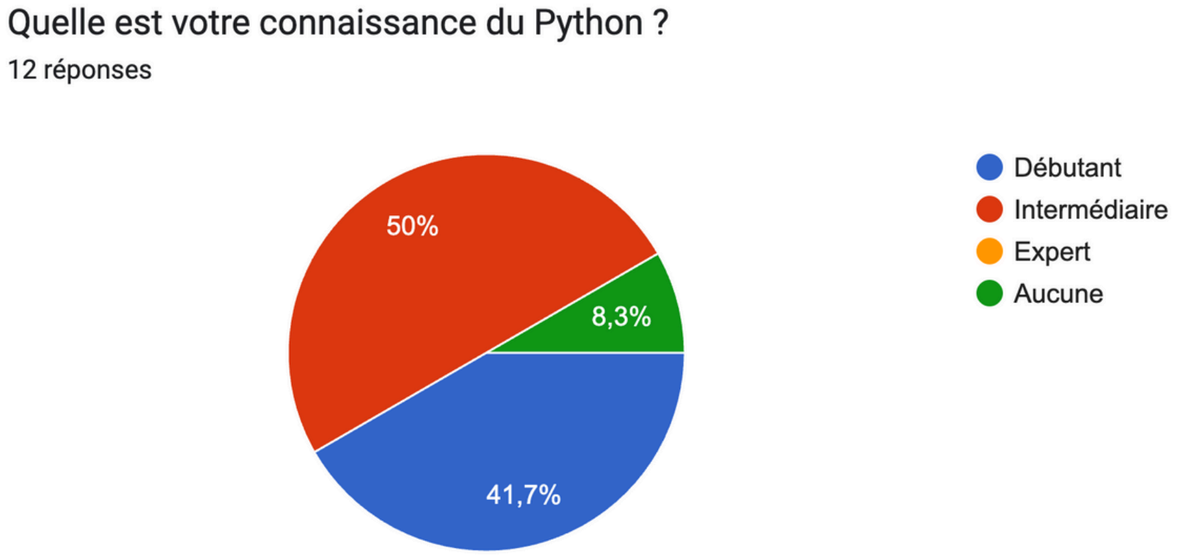
Discussion

ATELIER D'AUJOURD'HUI...

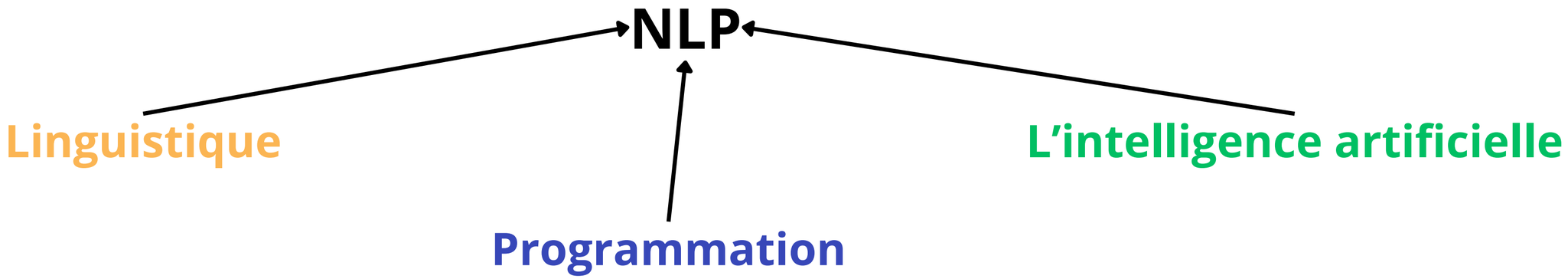
- **Pour qui ?**
  - Débutants en NLP avec bases en Python
  - Pas besoin de connaissances en maths

- **Ce que vous allez apprendre**
  - Utiliser spaCy
  - Structurer un projet NLP
  - Intégrer des outils modernes

- **Pré-requis techniques**
  - Maîtrise de la syntaxe Python
  - Environnements virtuels
  - Jupyter Notebook



TRAITEMENT AUTOMATIQUE DU LANGAGE (NLP)



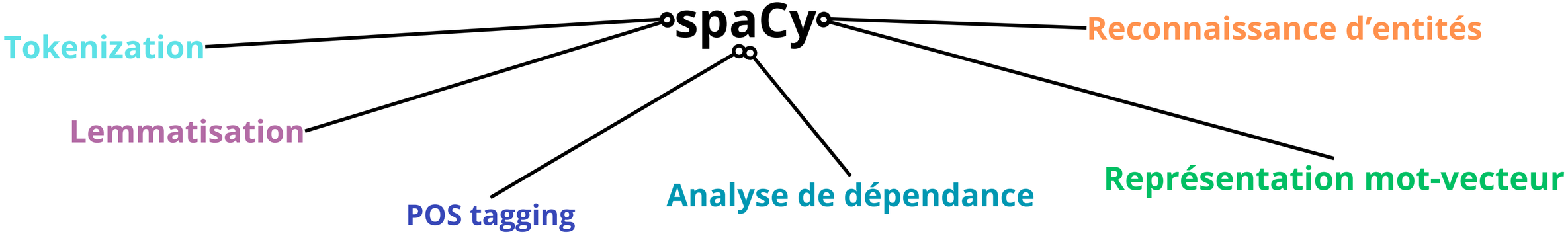
Applications du NLP

- Branche de l’IA dédiée aux interactions entre ordinateurs et langage humain
- Analyse et compréhension automatique des textes
- Utile pour traiter de grands volumes de données non structurées (réseaux sociaux, avis, etc.)

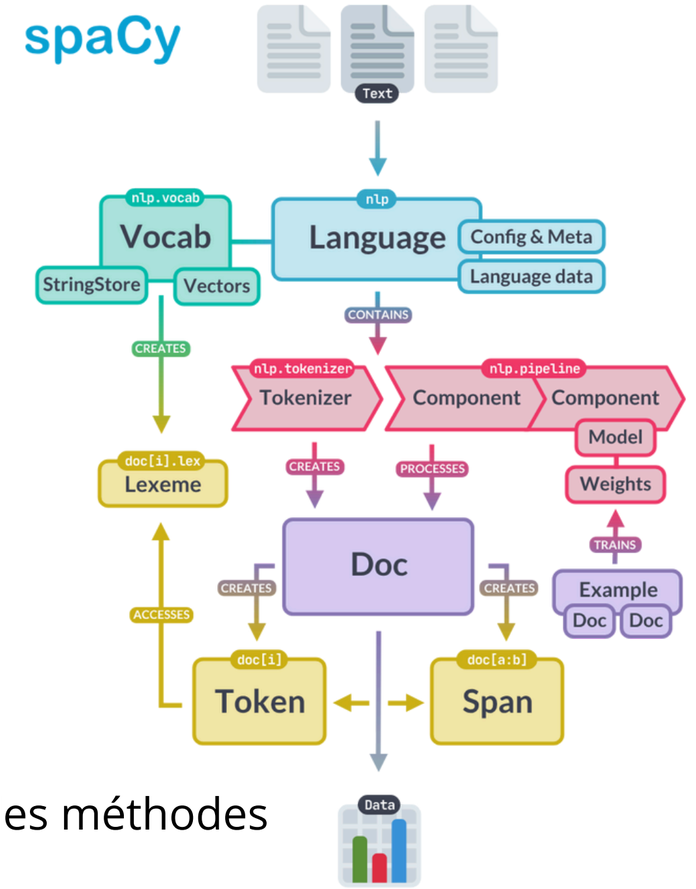
QU'EST-CE QUE SPACY ?

- Bibliothèque Python open source pour le Traitement du Langage Naturel
- Écrite en Cython, conçue pour la production
- API simple, rapide et efficace
- Développée par Explosion AI
  → spaCy = le NumPy du NLP

FONCTIONNALITÉS CLÉS DE SPACY



LIBRARY ARCHITECTURE



- Python est un langage orienté objet (**POO**) :

Il manipule des objets qui combinent données et méthodes.

- spaCy est une bibliothèque basée sur des objets spécifiques au NLP : comme Language, Doc, Token, etc.

- Un objet : une instance de classe qui contient des attributs (données) et des méthodes (comportements).

JUPITER NOTEBOOK

SPACY VS NLTK

🔹 spaCy
- Intègre directement des algorithmes optimisés
- Utilise des modèles statistiques modernes
- Approche orientée objet (mots = objets)
- Excellente performance en tokenization et POS tagging

🔹 NLTK
- Propose un choix d’algorithmes plus large
- Prend en charge plus de langues
- Approche ligne par ligne (code brut)
- Meilleure tokenization de phrases

👉 Deux bibliothèques puissantes, avec des usages complémentaires

CONCLUSION : CE QUE NOUS AVONS ACCOMPLI AUJOURD'HUI

• Compréhension du pipeline spaCy avec spacy.blank()
→ Tokénisation et segmentation de phrases

• Analyse d'un texte littéraire (Voyage au bout de la nuit)

• Utilisation de modèles statistiques avec spacy.load()
→ Lemmatisation, POS, NER

• Applications avancées :
→ Visualisation grammaticale (DisplaCy)
→ Règles avec le Matcher (ex : passé composé)
→ Similarité sémantique (word embeddings)

DISCUSSION ?


OBTIC
Observatoire des textes, des idées et des corpus
LETTRES SORBONNE UNIVERSITÉ
scai
BnF Bibliothèque nationale de France
Mikhail Biriuchinskii (He/Him)
NLP Developer | Computational linguist
Paris, Île-de-France, France · Contact info
500+ connections
Sorbonne Université
Université Paris Nanterre