

**Demandez et votre  
application **répondra**, la  
**recherche sémantique**  
avec **PHP** et **Meilisearch****

**Le bonsoir**

- ▶ Guillaume Loulier / @Guikingone
- ▶ Casse des CI @SensioLabs
- ▶ Travaille majoritairement avec PHP et Rust, un brin de Zig (merci ZML)
- ▶ Publie une newsletter hebdomadaire parlant de cloud, machine learning, PHP et bien plus via [Substack](#)



01

# Planifions

**01 - De quoi parlons-nous**

**02 - Un ordinateur n'apprend pas, il stocke**

**03 - Et en situation réelle ?**

**04 - Des conseils ?**

**Parlons peu,  
parlons  
sérieusement**

# L'heure des comptes

# Chercher n'est pas **aisé**

---

- La majorité des “expériences” de recherche sont bâties autour de la recherche par **mot-clef**
- La recherche par mot-clefs est basé sur **l'occurrence** d'un mot
- Si un mot-clef apparait **X** fois, le document retourné est-il **pertinent** ? Pourquoi ?



# Tout a un sens, enfin ...

---

- ▶ Le cerveau humain fonctionne par motifs, faits, expérimentations et habitudes, un ordinateur se limite aux binaires et mathématiques
- ▶ Un ordinateur ne peut pas chercher de façon logique / sensée
- ▶ Les mathématiques peuvent-ils nous aider ? Comment ?





# Retour en 6ème

---

- ▶ Que sont les mots et phrases si ce n'est des **séquences d'entiers** issues de l'alphabet ?
- ▶ Si tout peut se résumer à des entiers, peut-on effectuer des **opérations** dessus et trouver des **récurrentes** ?
- ▶ Pourrait-on voir ces séquences dans un environnement en **trois dimensions** ? Quid des **images** ?



*Oui mais l’“IA” ?*

# L'IA n'existe pas

---

- L'intelligence générale est un **mythe**, nous sommes tous(tes) **spécialisé(e)s**
- L'IA ? Un **buzzword** de marketeux pour vendre des apps utilisant OpenAI
- Le fond du sujet est la **transformation d'énergie** et sa **conservation**, sujet complexe et pour une autre fois



# Who run the world? Energy

---

Les révolutions récentes sont centrées

- ▶ autour des **LLMs / transformers**, un *sous-type* de “réseau neuronal”

Tout repose sur l'idée de *comprendre* et *analyser* un **contexte**

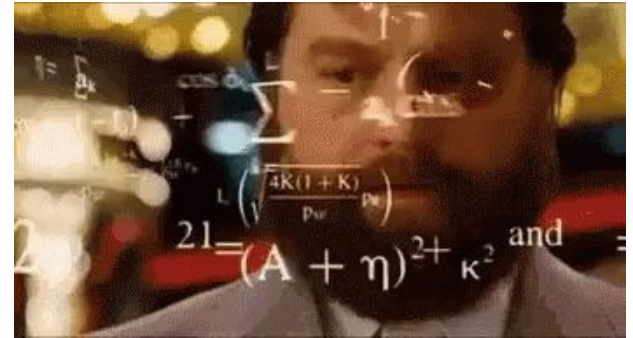
Les LLMs ne sont pas **intelligents**, ils

- ▶ excellent avec les **probabilités** et la **tokenization**, ils sont tous **biaisés**



La différence  
est une  
similarité

- La recherche sémantique (ou **recherche vectorielle**) est centrée autour de la proximité et de la similarité
- Plus les nombres sont **proches**, plus ils semblent **similaires**, plus la **pertinence** semble être correcte
- Quid de l'**intention** ? Quid du **sens** ?



# Déduire n'est pas percevoir

---

Le chien court après la balle de tennis dans la cuisine

---

Le chien court après la balle de tennis dans la cuisine

---

La balle de tennis court après le chien dans la cuisine

---



# Rappel des faits

Le	chien	court	après	la	balle	de	tennis	dans	la	cuisine
0	-1	0	1	-1	0	1	1	1	0	-1
0	2	2	2	0	0	2	-1	0	2	2
3	0	0	2	3	3	0	1	0	-1	2
1	1	2	0	1	-1	3	2	1	0	1



Encoder



[-0.88440161, -0.00996133, 0.243678553, ...]

\* Les valeurs sont donnés à titre d'exemple



► **Similarité** ne veut pas dire **pertinence** ou **intention**

↳ **0.1** est proche de **0.2** mais aussi de **0**, que cache **0** ?

↳ La similarité ne veut pas dire que la **signification** est bonne

↳ Selon le contexte, la similarité peut introduire un **biais**

↳ Le **contexte** est la clef de voûte, tout le reste n'est que du **bruit**

↳ Plus le contexte est **détaillé**, plus les résultats seront **cohérents**

↳ “Le bonsoir” est proche de “bonsoir” mais “bonjour” est plus pertinent selon l'**intention**

# Meilisearch

# Usain sans Bolt

---

- ▶ Créé pour un besoin interne chez **LVMH**
- ▶ Première itération en **Go**, réécrit en **Rust**, rapide (temps de réponse < ~ **50ms**)
- ▶ **Open-source** (MIT) et conçu en **France**
- ▶ SDK **PHP**, bundle **Symfony** et bien d'autres (JS, etc)



- ▶ Vector store natif
- ▶ Peut utiliser Ollama, OpenAI, Anthropic et autres (REST est la clef)
- ▶ API unifiée pour la recherche classique et sémantique
- ▶ Compatible avec la recherche fédérée, configuration au runtime



**Sortons la  
pelle**

► Construit via [symfony/demo](#)

► Disponible via <https://github.com/Guikingone/SyllusconSemantic>

# Concluons

# Contextualisons

---

- Plus précis vous serez, plus cohérents et *pertinents* seront les résultats
- Choisir le bon modèle est difficile, *expérimenter*, *optimiser* et *monitorer*
- Envie de faire *mieux* ? (Graph)*RAG* ?
- Ne tentez pas de *réinventer* la roue, les mathématiques restent les mathématiques





# Dernière chose

---

- ▶ Pensez aux **utilisateurs** plus qu'aux performances
- ▶ La recherche sémantique est une **idée**, pas une **promesse**
- ▶ Parfois, une recherche par mot-clef est **suffisante**
- ▶ Restez curieux, lisez des **retours d'expérience** et **testez**



# Des bisous

