

HAYSTACK



adelean  
EXTRACT TRANSFORM SEARCH

# Dive into NLP with the Elastic Stack



**Pietro Mele**  
Software Engineer  
@a2lean



**Lucian Precup**  
Principal Consultant  
@lucianprecup

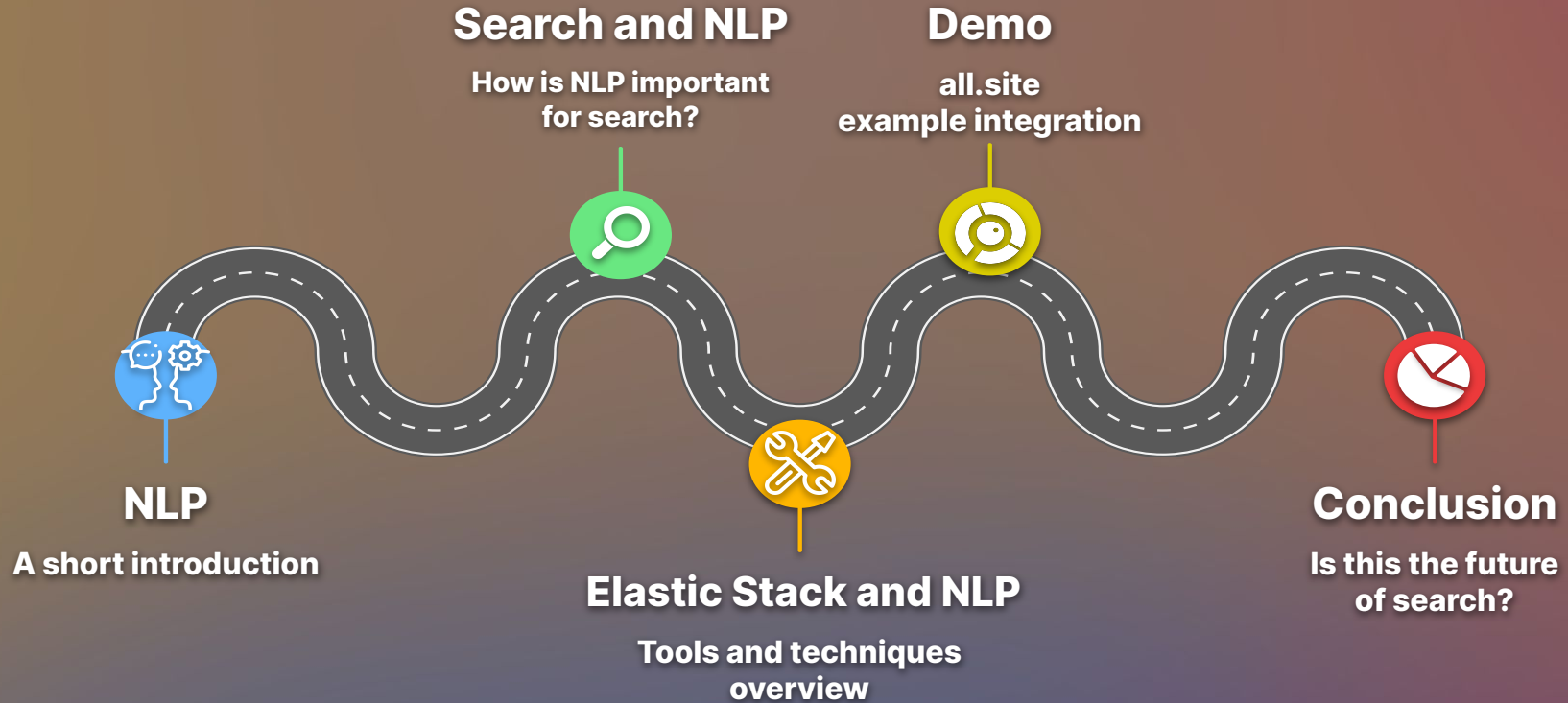
# Who are we ?

🔍 Adelean

- 🔍 Experts in **Search** technologies
- 🔍 Integrators of **Elasticsearch, OpenSearch** and **Solr**
- 🔍 **Consulting** and **Training** providers
- 🔍 Developers of **a2 - E-commerce** and **Enterprise Search** solution
- 🔍 Developers of **all.site** - your **Collaborative Search Engine**



# Our roadmap for today



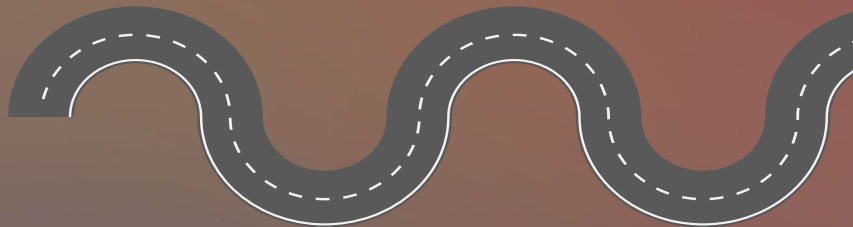
# What we leave for another talk

## NLP integration with OpenSearch

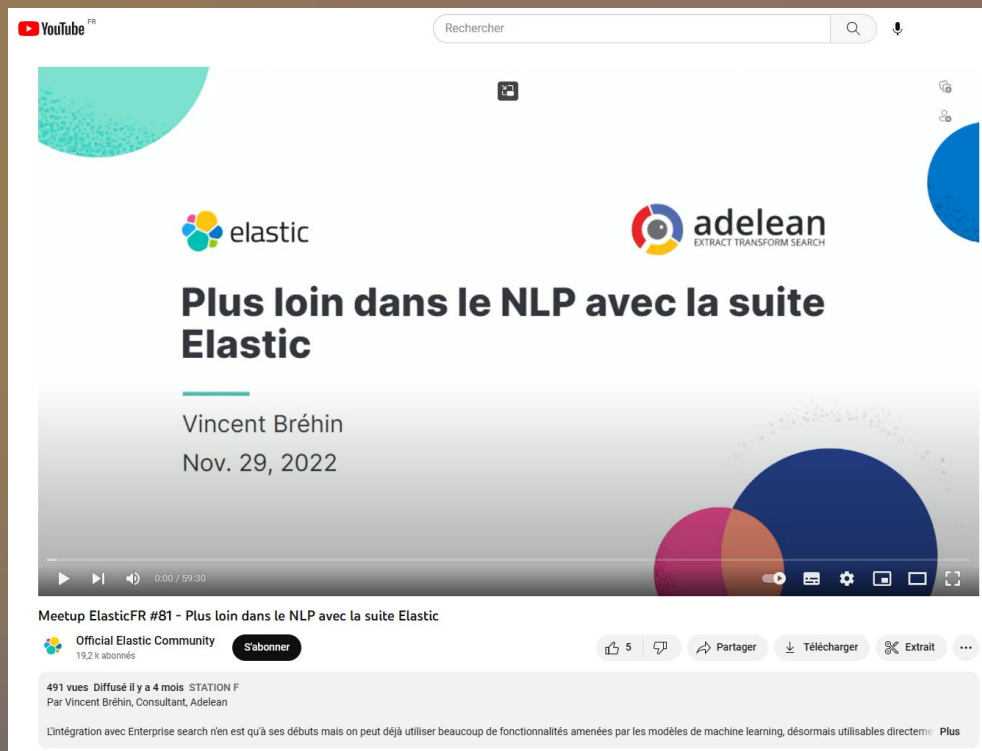
- the Neural Search plugin
- ...

## NLP and Solr

- Apache OpenNLP integration
- Machine Learning in Solr
- ...



# Credits for the initial presentation (in French)



The screenshot shows a YouTube video player interface. At the top left is the YouTube logo. A search bar with the text 'Rechercher' is at the top center. The video title is 'Meetup ElasticFR #81 - Plus loin dans le NLP avec la suite Elastic'. The channel name is 'Official Elastic Community' with 19.2k subscribers. The video is by Vincent Bréhin, dated Nov. 29, 2022. The video player shows a progress bar at 0:00 / 59:30. Below the player are interaction buttons: 5 likes, a comment icon, 'Partager', 'Télécharger', 'Extrait', and a menu icon. The video description starts with 'L'intégration avec Enterprise search n'en est qu'à ses débuts mais on peut déjà utiliser beaucoup de fonctionnalités amenées par les modèles de machine learning, désormais utilisables directement.' Logos for 'elastic' and 'adelean EXTRACT TRANSFORM SEARCH' are visible in the video frame.

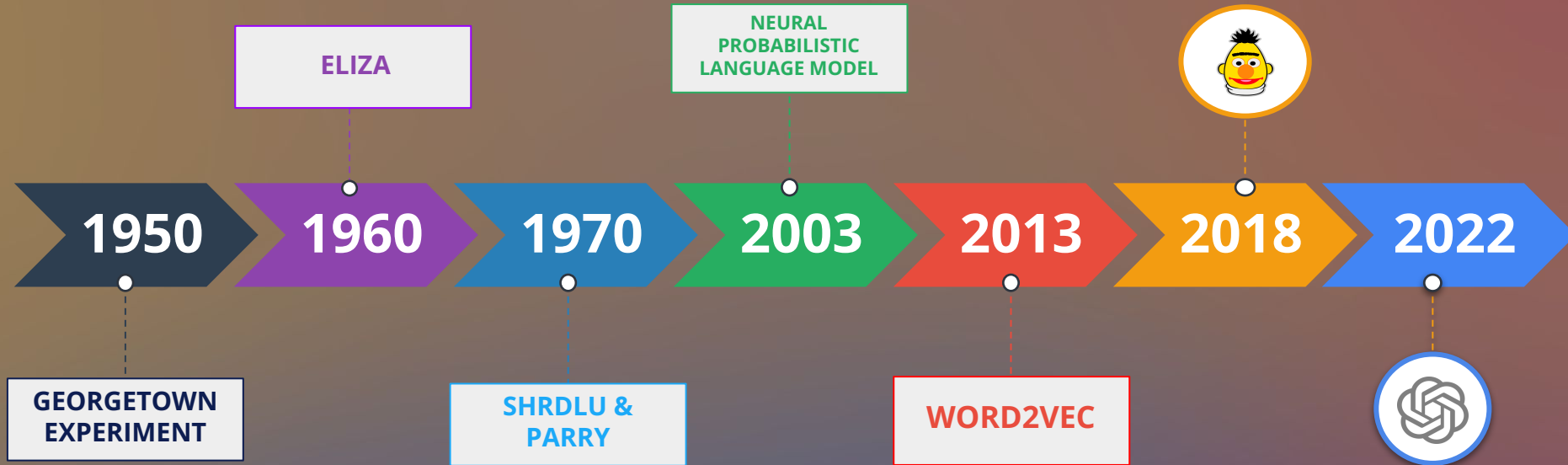


**ElasticFR Meetup #81 - Dive into NLP with the Elastic Stack**



# NLP

## Story of Natural language processing



**How did we get this far?**



**DATA**



**DATA EVERYWHERE**



**YES, BUT CAN IT RUN MINECRAFT?**

CHAPPIE

WORDS  
MEANING  
LANGUAGE  
CONTEXT  
SEMANTICS



DADDY HAPPY

NOT HAPPY = SAD



# ATTENTION IS ALL YOU NEED

PAPER DISSECTION

# NLP and Search

orange juice

When is the next school vacation in my area?

## NLP and Search: a long story...

- **Search queries are originally expressed in natural language**
- **Search systems need to interpret natural language**

## NLP has always been around:

- **Analyzers (tokenizers, stemmers, synonyms...)**
- **Inverted index**
- **Scoring (vector model)**

# NLP models and vector search: a paradigm shift

Move beyond text-matching



How fast should my internet be?

In order to stream from our service you will need a high quality connection. The required connection speed for using the service will vary depending on the quality of your internet connection. If you wish to stream in HD, we recommend a minimum of 5 Mbps. For most customers we recommend at least...

# What NLP ( and vector search ) allows you to do?

- **Monitor market trends**
- **Question answering**
- **Product similarity search**
- **Act on customer reviews**
- **Personalized recommendations**
- **Mine maintenance logs**
- **Query technicals manuals**
- **Identify similar cases**
- **Find reports of adverse drug effects**



# SPARSE

VS

# DENSE

PRO

Faster retrieval



Exact matching



Vocabulary mismatch



No semantic



YES to Semantic



Multimodal



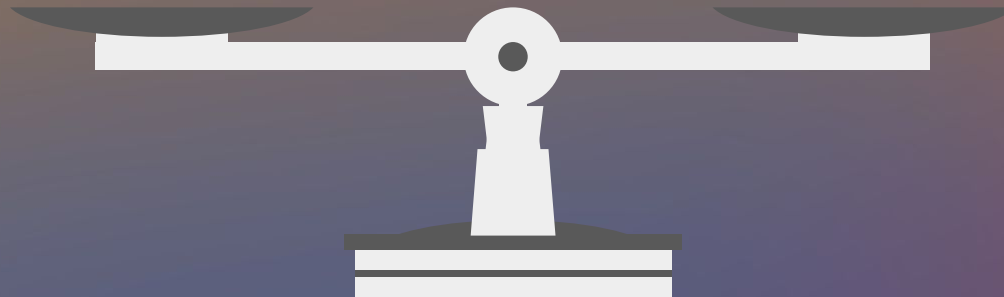
Model to fine-tune



Interpretability



CONS



# NLP tasks: you know, for search

## Extract information:

- **Fill-Mask**
- **Named Entity Recognition**
- **Summarization**
- **Question answering**

## Classify text:

- **Language identification**
- **Sentiment analysis & Binary Text Classification**
- **Zero-Shoot Classification**

## Search and compare text:

- **Text embedding**
- **Text similarity**



# NLP Tasks

## Named Entity Recognition (NER)

Haystack **MISC** will be held in Charlottesville **LOC** the week of 24th April and is organized by the team at OpenSource Connections **ORG**. Trey Grainger **PER** is speaking for the opening keynote. Haystack **MISC** is the conference for improving search relevance.

# NLP Tasks

## Sentiment Analysis



My experience  
so far has been  
fantastic!

POSITIVE



The product is  
ok I guess

NEUTRAL



Your support team  
is useless

NEGATIVE

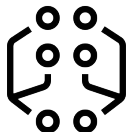
# NLP Tasks

## Zero-shoot Classification

Classifying from previously unseen classes ( labels )

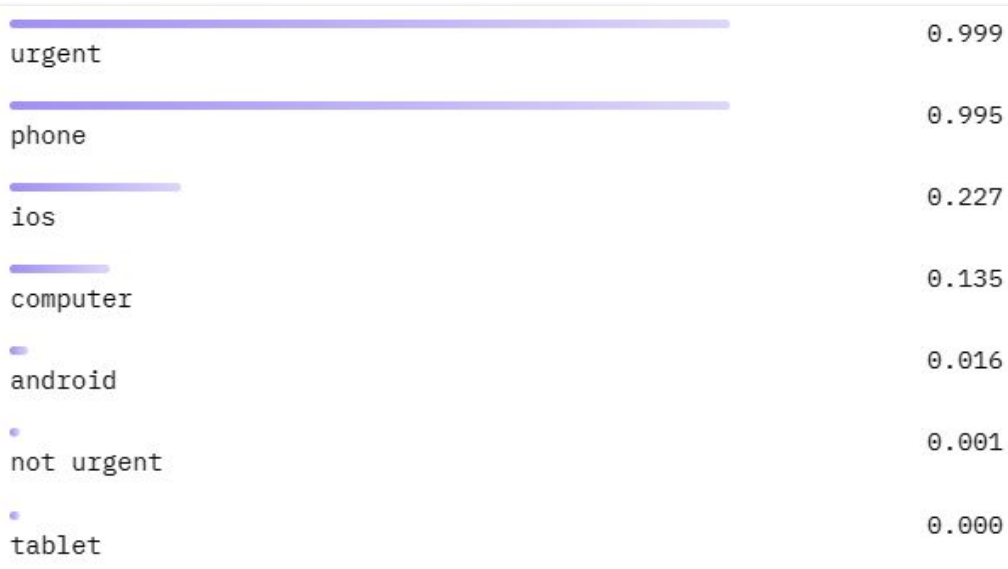
### Input:

"I have a problem with my  
iphone that needs to be  
resolved asap!!"



### Possible classes

Urgent, phone, ios,  
computer, android, not  
urgent, tablet



# NLP Tasks

## Summarization

### Natural Language Processing

Natural language processing (NLP) is a subfield of linguistics, computer science, and artificial intelligence concerned with the interactions between computers and human language, in particular how to program computers to process and analyze large amounts of natural language data. The result is a computer capable of "understanding" the contents of documents, including the contextual nuances of the language within them. The technology can then accurately extract information and insights contained in the documents as well as categorize and organize the documents themselves.

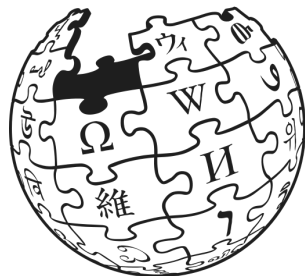


### Natural Language Processing

Natural language processing (NLP) is a subfield of linguistics, computer science, and artificial intelligence concerned with the interactions between computers and human language, in particular how to program computers to process and analyze large amounts of natural language data.

# NLP Tasks

## Question-answering



### Context:

“The Amazon rainforest, also called **Amazon jungle or Amazonia**, is a moist broadleaf tropical rainforest in the Amazon biome that covers most of the Amazon basin of South America. This basin encompasses 7,000,000 km<sup>2</sup> (2,700,000 sq mi), of which 5,500,000 km<sup>2</sup> (2,100,000 sq mi) are covered by the rainforest. This region includes territory belonging to **nine** nations and 3,344 formally acknowledged indigenous territories.”

**Question:** What other name is used to describe the Amazon rainforest?

**Answer:** **the Amazon Jungle or Amazonia**

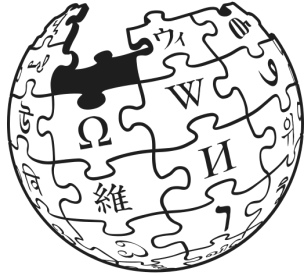
**Question:** How many countries the Amazon forest cover?

**Answer:** **nine**

# NLP Task

## Generative Question Answering

### Context:



“The Amazon rainforest, also called Amazon jungle or Amazonia, is a moist broadleaf tropical rainforest in the Amazon biome that covers most of the Amazon basin of South America. This basin encompasses 7,000,000 km<sup>2</sup> (2,700,000 sq mi), of which 5,500,000 km<sup>2</sup> (2,100,000 sq mi) are covered by the rainforest. This region includes territory belonging to nine nations and 3,344 formally acknowledged indigenous territories. The majority of the forest is contained within Brazil, with 60% of the rainforest, followed by Peru with 13%, Colombia with 10%, and with minor amounts in Venezuela, Ecuador, Bolivia, Guyana, Suriname and French Guiana.”

**Question:** How many countries the Amazon forest cover?

**Answer:** The Amazon rainforest covers 9 countries. It covers parts of the following countries: Bolivia, Brazil, Colombia, Ecuador, Guyana, Peru, French Guiana, Suriname and Venezuela.

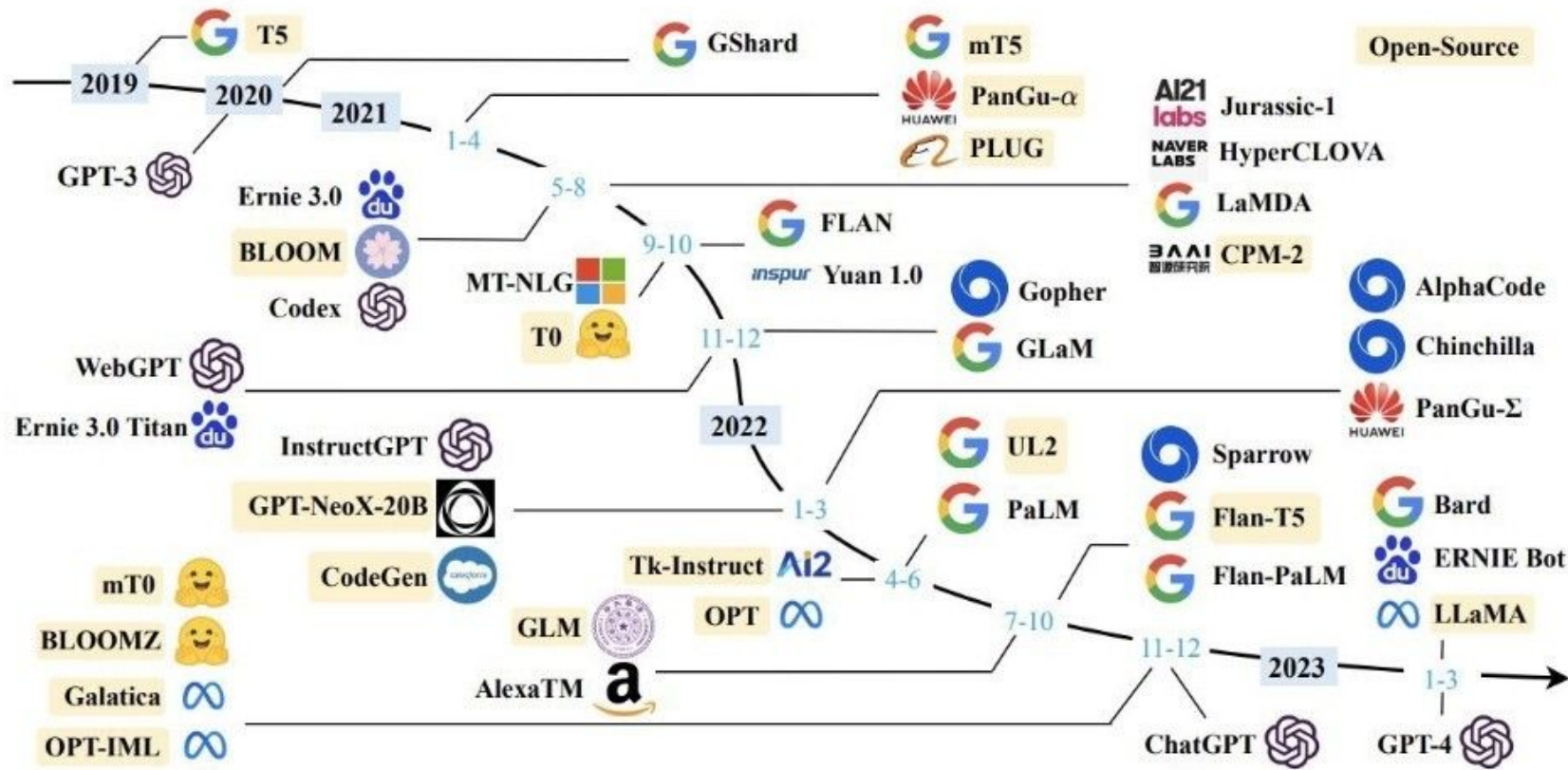
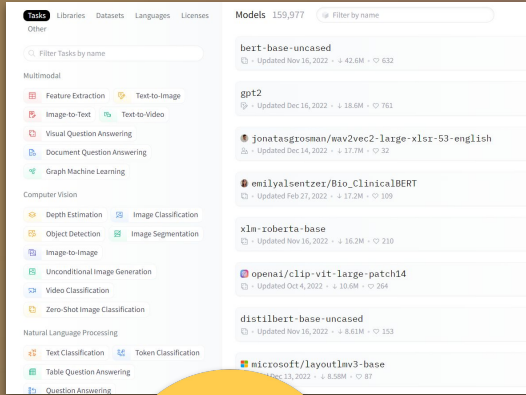
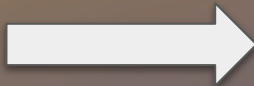
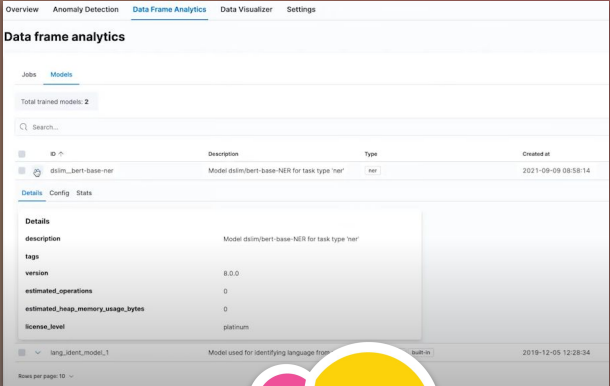


Fig. 1. A timeline of existing large language models (having a size larger than 10B) in recent years. We mark the open-source LLMs in yellow color.

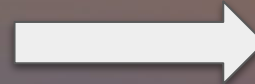
# Model preparation & upload



```
$eland_import_hub_model  
<authentication> \  
--url http://localhost:9200/ \  
--hub-model-id  
elastic/distilbert-base-cased-finetuned  
--task-type ner \  
--start
```



eland





# Hugging Face: overview

Different Tasks

Tasks Libraries Datasets Languages Licenses Other

Filter Tasks by name

Multimodal

- Feature Extraction
- Text-to-Image
- Image-to-Text
- Text-to-Video
- Visual Question Answering
- Document Question Answering
- Graph Machine Learning

Computer Vision

- Depth Estimation
- Image Classification
- Object Detection
- Image Segmentation
- Image-to-Image
- Unconditional Image Generation
- Video Classification
- Image-to-Video
- Image Classification

**NLP Tasks**

Natural Language Processing

- Text Classification
- Token Classification
- Table Question Answering
- Question Answering
- Zero-Shot Classification
- Translation
- Summarization
- Conversational
- Text Generation
- Text2Text Generation
- Fill-Mask
- Sentence Similarity

Audio

- Text-to-Speech
- Automatic Speech Recognition
- Audio-to-Audio
- Audio Classification

Models 168,998 Filter by name

new Full-text search Sort: Most Downloads

- bert-base-uncased**  
Updated Nov 16, 2022 · ↓ 41.1M · ♥ 671
- jonatasgrosman/wav2vec2-large-xlsr-53-english**  
Updated 10 days ago · ↓ 31.3M · ♥ 51
- gpt2**  
Updated Dec 16, 2022 · ↓ 19.8M · ♥ 837
- xlm-roberta-base**  
Updated Nov 16, 2022 · ↓ 17.1M · ♥ 227
- Davlan/distilbert-base-multilingual-cased-ner-hrl**  
Updated Jun 27, 2022 · ↓ 14.8M · ♥ 15
- openai/clip-vit-large-patch14**  
Updated Oct 4, 2022 · ↓ 10.2M · ♥ 299
- microsoft/layoutlmv3-base**  
Updated Dec 13, 2022 · ↓ 8.27M · ♥ 101
- emilyalsentzer/Bio\_ClinicalBERT**  
Updated 4 days ago · ↓ 8.21M · ♥ 119
- distilbert-base-uncased**  
Updated Nov 16, 2022 · ↓ 7.95M · ♥ 163
- distilroberta-base**  
Updated Nov 17, 2022 · ↓ 6.91M · ♥ 52
- roberta-base**  
Updated 29 days ago · ↓ 6.45M · ♥ 143
- xlm-roberta-large**  
Updated 11 days ago · ↓ 6.35M · ♥ 120
- bert-base-cased**  
Updated Nov 16, 2022 · ↓ 6.27M · ♥ 87
- t5-base**  
Updated Jan 24 · ↓ 6.23M · ♥ 167
- openai/clip-vit-base-patch32**  
Updated Oct 4, 2022 · ↓ 5.78M · ♥ 144
- albert-base-v2**  
Updated Aug 30, 2021 · ↓ 3.48M · ♥ 47
- runwayml/stable-diffusion-v1-5**  
Updated Jan 27 · ↓ 3.26M · ♥ 6.51k
- bert-base-multilingual-cased**  
Updated Nov 17, 2022 · ↓ 2.81M · ♥ 123
- google/electra-base-discriminator**
- facebook/bart-large-mnli**

Pre-trained models

# Model management and allocation

The screenshot shows the Elastic ML Model Management interface. The top navigation bar includes 'Overview', 'Anomaly Detection', 'Data Frame Analytics', 'Model Management', 'Data Visualizer', and 'Settings'. The 'Model Management' section is active, displaying 'Trained Models' with an 'EXPERIMENTAL' tag and a 'Refresh' button. Below this, there are tabs for 'Models' and 'Nodes'. A search bar is present above a table of models. The table lists 7 models with columns for ID, Description, and Type.

ID	Description	Type
distilbert-base-uncased-finetuned-sst-2-english	Model distilbert-base-uncased-finetuned-sst-2-english for task type 'text_classification'	pytorch   text_classification
dsim_bert-base-ner	Model dsim/bert-base-NER for task type 'ner'	pytorch   ner
elastic_distilbert-base-cased-finetuned-conll03-english	Model elastic/distilbert-base-cased-finetuned-conll03-english for task type 'ner'	pytorch   ner
lang_ident_model_1	Model used for identifying language from arbitrary input text.	lang_ident   classification   built-in
sentence-transformers_clip-vit-b-32-multilingual-v1	Model sentence-transformers/clip-vit-b-32-multilingual-v1 for task type 'text_embedding'	pytorch   text_embedding
sentence-transformers_msmarco-minilm-l12-v3	Model sentence-transformers/msmarco-MiniLM-L12-v3 for task type 'text_embedding'	pytorch   text_embedding
typeform_distilbert-base-uncased-mnli	Model typeform/distilbert-base-uncased-mnli for task type 'zero_shot_classification'	pytorch   zero_shot_classification

This screenshot shows the details view for a specific model in the Elastic ML Model Management interface. The 'Model Management' section is active, and the 'Details' tab is selected. The model name is 'distilbert-base-uncased-finetuned-sst-2-english'. Below the model name, there are tabs for 'Details', 'Config', 'Stats', and 'Pipelines'. The 'Stats' tab is active, showing 'Ingest stats' for the model.

Processor type	Count	Current
inference	13750	0
sst	12745	0

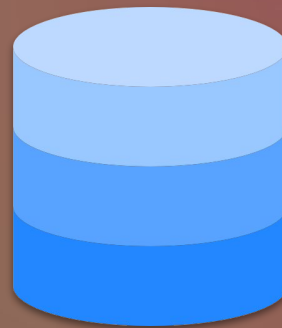
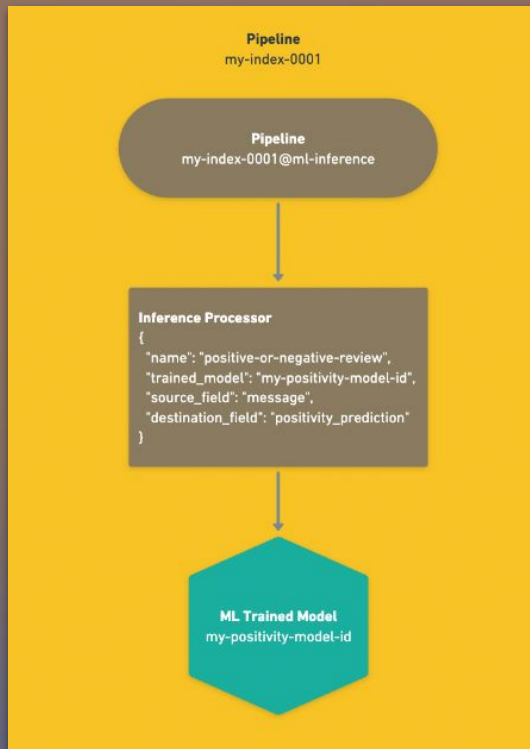
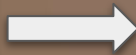
This screenshot shows the 'Nodes' view in the Elastic ML Model Management interface. The 'Nodes' tab is active, displaying 'Total machine learning nodes: 1'. Below this, there is a search bar and a table of nodes. The table has columns for 'Name', 'Total memory', and 'Memory usage'.

Name	Total memory	Memory usage
instance-000000002	64GB	<div style="width: 10%;"></div>

# Inference at ingest

## Document:

```
POST my-index-0001/_doc?pipeline=my-index-0001
{
  "@timestamp": "2099-11-15T13:12:00",
  "message": "This is the best feature!",
  "_run_ml_inference": true
}
```



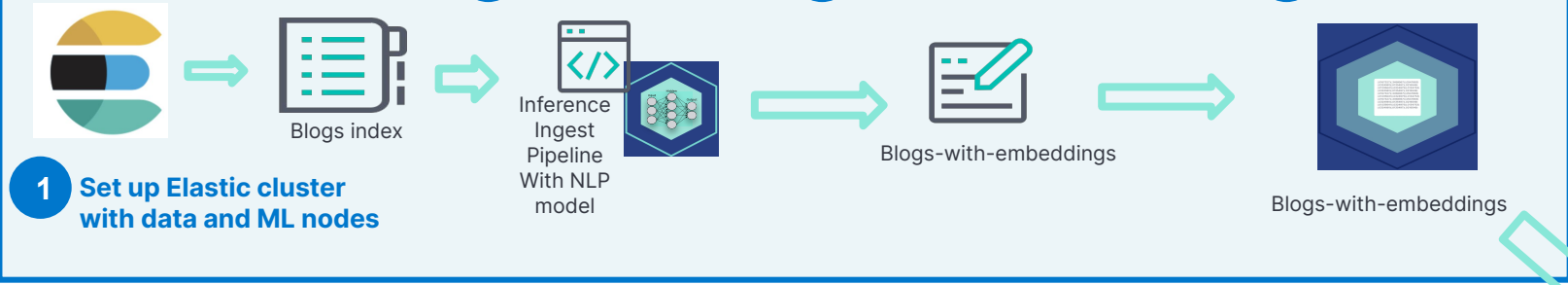
## Enriched document:

```
{
  "@timestamp": "2099-11-15T13:12:00",
  "message": "This is the best feature!",
  "positivity_prediction": "0.91644"
}
```

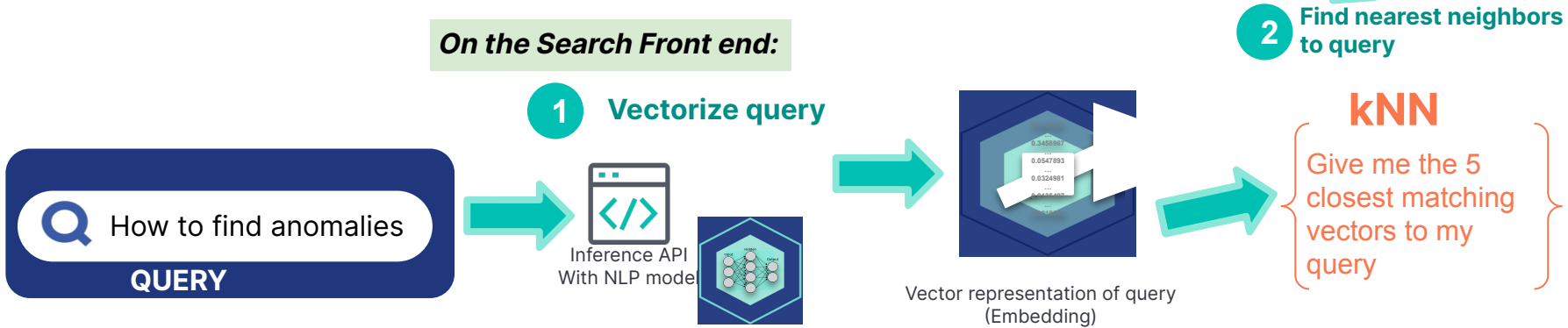


# Workflow semantic search

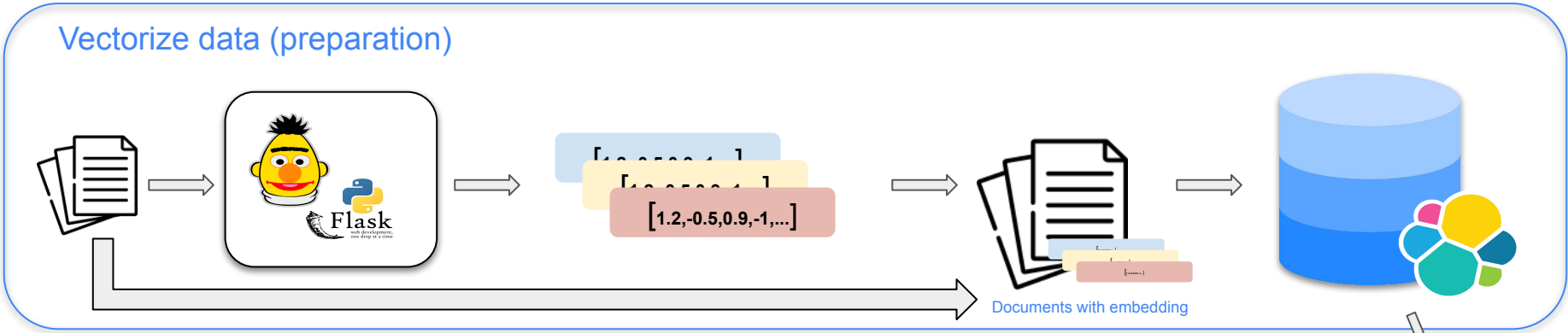
## Vectorize data (prep):



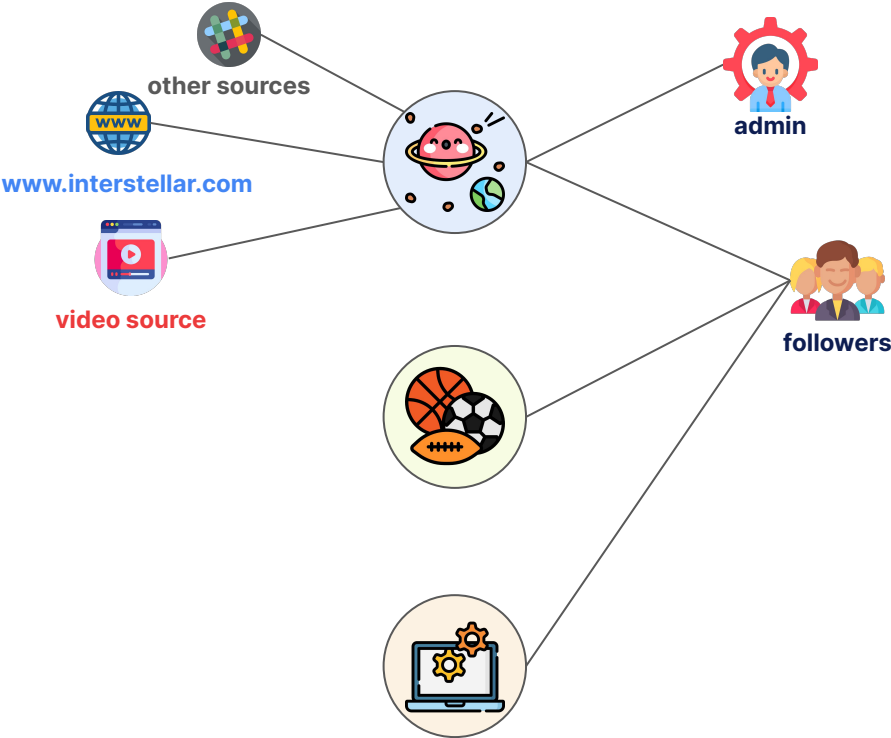
## On the Search Front end:



# Workflow for semantic search - home made

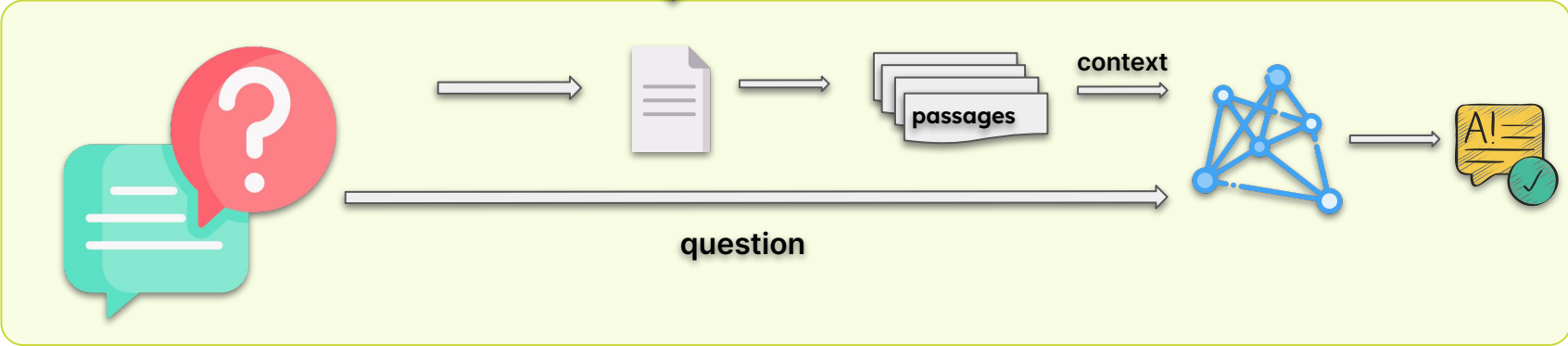


# all.site: your collaborative search engine



- ✓ Collaborative, as Wikipedia
- ✓ Relevant, as Reddit
- ✓ Community driven, as GitHub

# Question answering workflow





# Question answering implementation in real life

```
model_name = "deepset/tinyroberta-squad2"
```

```
tokenizer = AutoTokenizer.from_pretrained(model_name)  
model = ORTModelForQuestionAnswering.from_pretrained(model_name, from_transformers=True)  
nlp = pipeline("question-answering", model=model, tokenizer=tokenizer)
```

☰ Connaître le monde

## Sources indexées

TYPE	URL	STATUS	DOCS
 WEB	<a href="https://www.lumni.fr/">https://www.lumni.fr/</a>	<span style="color: blue;">●</span> STARTED	17
 WEB	<a href="https://fr.wikipedia.org/">https://fr.wikipedia.org/</a>	<span style="color: blue;">●</span> STARTED	307

< 1 >

```
for hit in resp['hits']['hits']:  
    for fragment in hit['highlight']['attachment.content']:  
        context = cleanAndTrim(fragment)  
        QA_input = {'question': question,  
                    'context': context}  
  
        answer = nlp(QA_input)  
  
        if (answer.get('score') > 0.50):  
            return answer.get('answer')  
  
return "I can't answer..."
```



# Question answering use case

The screenshot displays a user interface for a question answering system. At the top left, the text 'a// // Demo QA' is visible. A search bar contains the query 'Quelle est la capitale européenne?'. To the right of the search bar are icons for language (en), home, user profile, and share. Below the search bar, there is a 'Results sorted by:' dropdown menu and a '2 Results Relevance' indicator. A 'Source' dropdown menu is open, showing two options: 'fr.wikipedia.org 1' and 'www.lumni.fr 1'. The main content area shows a detailed answer snippet for the query. The snippet includes the text 'Answer: La ville de Bruxelles' with a pin icon, followed by a 'Fragment' of text: 'La ville de Bruxelles est souvent considérée comme la seule capitale de l'Europe, à tort. ... : le moteur de l'Europe Où est la capitale de l'Europe ? ... oui non Fermer la fenêtre d'ajout au favoris et voir plus tardOù est la capitale de l'Europe ?'. Below this, a video result is shown with a globe icon, the title 'Où est la capitale de l'Europe ? - Vidéo Histoire | Lumni |', and a description: 'C Jamy, les extraitsLa ville de Bruxelles est souvent considérée comme la seule capitale de l'Europe, à tort. ... : le moteur de l'Europe Où est la capitale de l'Europe ? ... oui non Fermer la fenêtre d'ajout au favoris et voir plus tardOù est la capitale de l'Europe ?'.

DEMO@TIME

# Language classification

and overview of Trained Model interface

## Trained Models

Auto refresh Off Refresh

Total trained models: 4

Search... Type ▾

ID	Description	Type ↓	State	Created at	Actions
<input type="checkbox"/> > sentence-transformers_all-minilm-l6-v2	Model sentence-transformers/all-MiniLM-L6-v2 for task type 'text_embedding'	pytorch text_embedding	started	Apr 24, 2023 @ 23:45:25.634	...
<input type="checkbox"/> > deepset_roberta-base-squad2	Model deepset/roberta-base-squad2 for task type 'question_answering'	pytorch question_answering	started	Apr 24, 2023 @ 23:34:34.034	...
<input checked="" type="checkbox"/> > dslim_bert-base-ner	Model dslim/bert-base-NER for task type 'ner'	pytorch ner	started	Apr 24, 2023 @ 16:50:10.140	...
<input checked="" type="checkbox"/> > lang_ident_model_1	Model used for identifying language from arbitrary input text.	lang_ident built-in classification		Dec 5, 2019 @ 13:28:34.594	🔗

Rows per page: 10 < 1 >

## Test trained model

lang\_ident\_model\_1

### Language identification

Test how well the model identifies the language of your text.

#### Input text

We are having a lot of fun at the Haystack in Charlottesville

Test

Output Raw output

We are having a lot of fun at the Haystack in Charlottesville

#### This looks like English

en	0.998
it	0.000617
fil	0.00052
ja	0.000397

We can test our model directly into the Trained Model interface of Kibana

# Named Entity Recognition: preparation

```
PUT _ingest/pipeline/infer_ner
```

```
{  
  "description": "Pipeline for ner inference",  
  "processors": [  
    {  
      "inference": {  
        "model_id": "dslim_bert-base-ner", A  
        "field_map": {  
          "text": "text_field" B  
        },  
        "target_field": "ner" C  
      }  
    }  
  ]  
}
```

1

```
POST _reindex
```

```
{  
  "source": {  
    "index": "ml_demo"  
  },  
  "dest": {  
    "index": "ml_demo_ner",  
    "pipeline": "infer_ner"  
  }  
}
```

2

1. Creating an ingest pipeline
  - a. Using the model\_id of the previously imported model for named entity recognition
  - b. Definition of the source field for the inference
  - c. Definition of the destination field
2. Ingest through the pipeline you created

# Named Entity Recognition: showing results

```
  "ner": {  
    "predicted_value": ""[Dante Alighieri](PER&Dante+Alighieri) was an [Italian]  
      (MISC&Italian) poet, writer and philosopher. He died in [Ravenna](LOC&Ravenna), aged  
      56, on 14 September 1321. [Dante](PER&Dante) was born in [Florence](LOC&Florence),  
      [Republic of](LOC&Republic-of) [Florence](LOC&Florence), in what is now [Italy]  
      (LOC&Italy) around 1265. He is considered the "father" of the [Italian](MISC&Italian)  
      language and the "Supreme Poet" of [Italy](LOC&Italy). His depictions of [Hell]  
      (LOC&Hell), [Purgatory](LOC&Purgatory) and [Heaven](LOC&Heaven) provided inspiration  
      for [Western](MISC&Western) art and literature.[Dante](PER&Dante) was banished from  
      [Florence](LOC&Florence) in 1311. He lived in [Verona](LOC&Verona), [Sarzana]  
      (LOC&Sarzana), [Lucca](LOC&Lucca), and in [Cangrande della Scala](LOC&Cangrande+della  
      +Scala). He wrote [De Monarchia](MISC&De+Monarchia), proposing a universal monarchy  
      under [Henry](PER&Henry) VII. [Dante](PER&Dante)'s [Inferno](MISC&Inferno) was  
      published by 1317.[Dante](PER&Dante) was one of the first in [Roman Catholic Western]  
      (MISC&Roman+Catholic+Western) [Europe](LOC&Europe) to publish in the vernacular  
      language. He aimed to reach a wider audience, including laymen, clergymen and other  
      poets. His works include the [Divine Comedy](MISC&Divine+Comedy), [Convivio]  
      (MISC&Convivio), [The Banquet](MISC&The+Banquet), [La Vita Nuova](ORG&La+Vita+Nuova),  
      [La Quaestio de aqua et terra](ORG&La+Quaestio+de+aqua+et+terra), [Divina Commedia]  
      (MISC&Divina+Commedia), and [The Rime for the Land](MISC&The+Rime+for+the+Land  
      ).[Florence](LOC&Florence) hosted a re-trial of [Dante Alighieri](PER&Dante+Alighieri  
      ), who was banished in 1302. [Dante](PER&Dante)'s "[The Way of Beauty in](MISC&The  
      +Way+of+Beauty+in) [Dante](LOC&Dante)" is updated on [Dappledthings]  
      (ORG&Dappledthings).org on 28 August 2022. [Dante](PER&Dante) at the [Encyclopædia  
      Britannica](ORG&Encyclopædia+Britannica) is an [Italian](MISC&Italian) icon.""",
```

Full predicted value: different entities have been correctly identified

```
{  
  "entity": "Italian",  
  "class_name": "MISC",  
  "class_probability": 0.9996668600309809,  
  "start_pos": 23,  
  "end_pos": 30  
},  
{  
  "entity": "Ravenna",  
  "class_name": "LOC",  
  "class_probability": 0.5048188394330603,  
  "start_pos": 72,  
  "end_pos": 79  
},  
{  
  "entity": "Dante",  
  "class_name": "PER",  
  "class_probability": 0.9987073477838998,  
  "start_pos": 112,  
  "end_pos": 117  
},  
{  
  "entity": "Florence",  
  "class_name": "LOC",  
  "class_probability": 0.9996759975434635,  
  "start_pos": 130,  
  "end_pos": 138  
},
```

Entity View with class name, probability and offset - useful for faceting

# Question answering

```
POST _ml/trained_models/deepset__roberta-base-squad2/_infer 1
{
  "docs": [{ "text_field": "The Amazon rainforest (Portuguese: Floresta Amazônica or Amazônia; Spanish: Selva Amazónica, Amazonia or usually Amazonia; French: Forêt amazonienne; Dutch: Amazoneregenwoud), also known in English as Amazonia or the Amazon Jungle, is a moist broadleaf forest that covers most of the Amazon basin of South America. This basin encompasses 7,000,000 square kilometres (2,700,000 sq mi), of which 5,500,000 square kilometres (2,100,000 sq mi) are covered by the rainforest. This region includes territory belonging to nine nations. The majority of the forest is contained within Brazil, with 60% of the rainforest, followed by Peru with 13%, Colombia with 10%, and with minor amounts in Venezuela, Ecuador, Bolivia, Guyana, Suriname and French Guiana. States or departments in four nations contain \"Amazonas\" in their names. The Amazon represents over half of the planet's remaining rainforests, and comprises the largest and most biodiverse tract of tropical rainforest in the world, with an estimated 390 billion individual trees divided into 16,000 species."}],
  "inference_config": {
    "question_answering": {
      "question": "Which name is also used to describe the Amazon rainforest in English?"
    }
  }
}
```

```
"inference_results": [ 2
  {
    "predicted_value": "Amazonia or the Amazon Jungle", A
    "start_offset": 201, B
    "end_offset": 230,
    "prediction_probability": 0.7506235133118057 C
  }
]
```

1. Using the **\_infer** endpoint to infer an answer given a **context** (which can be a searched document, an extract of text, ...) and a **question**
2. Result showing:
  - a. answer
  - b. offset
  - c. probability prediction

# Vector Similarity - the journey

## Brute force vector similarity GA in 7.3

- Works great when filtering or reranking top
- Query latency on very large indices

## HNSW & KNN, in 8.0

- Bartering a little accuracy for a lot of scalability
- Better query latency on large scale indices

# Vector search: indexation

```
PUT ml_demo_vec
```

```
{
  "mappings": {
    "properties": {
      "text_field": {
        "type": "text"
      },
      "vector": {
        "type": "dense_vector",
        "dims": 384,
        "index": true,
        "similarity": "cosine"
      }
    }
  }
}
```

1

B

A

C

```
PUT _ingest/pipeline/infer_vector
```

```
{
  "description": "Pipeline for vectorization",
  "processors": [
    {
      "inference": {
        "model_id": "sentence-transformers_all-minilm-l6-v2",
        "target_field": "vector",
        "field_map": {
          "text": "text_field"
        }
      }
    }
  ]
}
```

2

A

B

C

1. Creating the destination index
  - a. Use `dense_vector` type for the field that will contain the vector
  - b. Remember to set `index` parameter to `true`
  - c. Define a similarity algorithm
2. Creating an ingest pipeline
  - a. Using the `model_id` of the previously imported model for vector embedding
  - b. Definition of the destination field
  - c. Definition of the source field for the inference
3. Reindex using the pipeline



# Vector search: the old fashioned brute-force

```
POST _ml/trained_models/sentence-transformers_all-minilm-l6-v2/_infer
{
  | "docs":[{"text_field": "Italian poet"}]
}
```

1

```
GET ml_demo/_search
{
  "query": {
    "script_score": {
      "query" : {
        | "match_all": {}
      },
      "script": {
        "source": "cosineSimilarity(params.query_vector, 'vector') + 1.0",
        "params": {
          | "query_vector": [ ]
        }
      }
    }
  }, "_source": ["text"]
}
```

2

```
"_id": "piFtQocBzRLictMKnx8w",
"_score": 1.9999999,
"_source": {
  | "text": ""Dante Alighieri was an Italian poet, writer
    and philosopher. He died in Ravenna, aged 56, on 14
    September 1321. Dante was born in Florence, Republic
    of Florence, in what is now Italy around 1265. He is
    considered the "father" of the Italian language and
    the "Supreme Poet" of Italy. His depictions of Hell,
    Purgatory and Heaven provided inspiration for
    Western art and literature.Dante was banished from
    Florence in 1311. He lived in Verona, Sarzana, Lucca
    , and in Cangrande della Scala. He wrote De
    Monarchia, proposing a universal monarchy under
    Henry VII. Dante's Inferno was published by 1317
    .Dante was one of the first in Roman Catholic
    Western Europe to publish in the vernacular language
    . He aimed to reach a wider audience, including
    laymen, clergymen and other poets. His works include
    the Divine Comedy, Convivio, The Banquet, La Vita
    Nuova, La Quaestio de aqua et terra, Divina Commedia
    , and The Rime for the Land.Florence hosted a re
    -trial of Dante Alighieri, who was banished in 1302.
    Dante's "The Way of Beauty in Dante" is updated on
    Dappledthings.org on 28 August 2022. Dante at the
    Encyclopædia Britannica is an Italian icon.""
}
```

# Vector search: `_search` endpoint and `approximate-knn`

```
GET ml_demo/_search
{
  "knn": {
    "field": "vector",
    "query_vector": [←→],
    "k": 1,
    "num_candidates": 100
  },
  "_source": [
    "id",
    "text"
  ],
  "explain": true
}
```

- The name of the vector field to search against. Must be a `dense_vector` field with indexing enabled.
- Query vector. Must have the same number of dimensions as the vector field you are searching against.
- Number of nearest neighbors to return as top hits.
- The number of nearest neighbor candidates to consider per shard.

The vector has to be calculated elsewhere, using the `_infer` endpoint

# Vector search: query\_vector\_builder feature

```
GET ml_demo/_search
{
  "knn": {
    "field": "vector",
    "query_vector_builder": {
      "text_embedding": {
        "model_id": "sentence-transformers__all-minilm-l6-v2",
        "model_text": "Italian writer"
      }
    },
    "k": 1,
    "num_candidates": 100
  },
  "_source": [
    "id",
    "text"
  ],
  "explain": true
}
```

- Available starting Elastic 8.7
- Text embedding directly in the query:
  - Model\_id to specify the model to use
  - Model\_text to indicate the text to embed
- No need to calculate the query vector elsewhere

# Vector search: combining keyword and vector search

```
GET ml_demo/_search
{
  "query": {
    "match": {
      "text": "Florence"
    }
  },
  "knn": {
    "field": "vector",
    "query_vector_builder": {
      "text_embedding": {
        "model_id": "sentence-transformers__all-minilm-l6-v2",
        "model_text": "Italian writer"
      }
    },
    "k": 1,
    "num_candidates": 100
  },
  "_source": [
    "id",
    "text"
  ],
  "explain": true
}
```

Query: using both match and knn

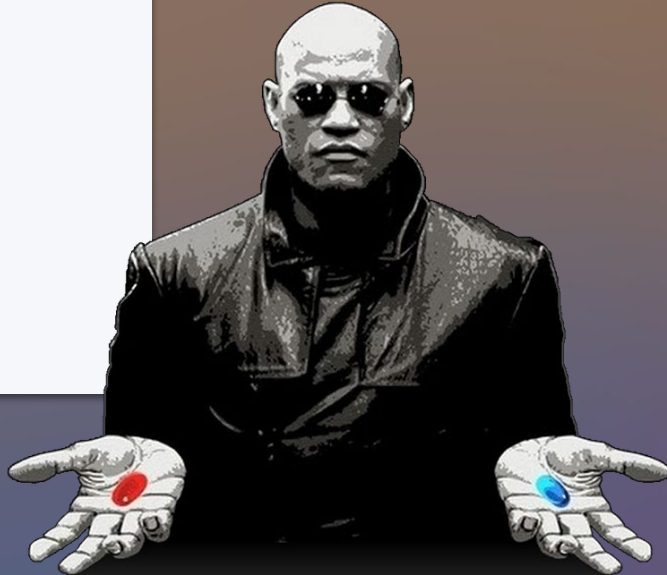
```
  "_id": "piFtQocBzRLictMKnx8w",
  "_score": 0.73311937,
  "_source": {
    "text": ""Dante Alighieri was an Italian poet, writer and philosopher. He died in Ravenna, aged 56, on 14 September 1321. Dante was born in Florence, Republic of Florence, in what is now Italy around 1265. He is considered the "father" of the Italian language and the "Supreme Poet" of Italy. His depictions of Hell, Purgatory and Heaven provided inspiration for Western art and literature. Dante was banished from Florence in 1311. He lived in Verona, Sarzana, Lucca, and in Cangrande della Scala. He wrote De Monarchia, proposing a universal monarchy under Henry VII. Dante's Inferno was published by 1317. Dante was one of the first in Roman Catholic Western Europe to publish in the vernacular language. He aimed to reach a wider audience, including laymen, clergymen and other poets. His works include the Divine Comedy, Convivio, The Banquet, La Vita Nuova, La Quaestio de aqua et terra, Divina Commedia, and The Rime for the Land. Florence hosted a re-trial of Dante Alighieri, who was banished in 1302. Dante's "The Way of Beauty in Dante" is updated on Dappledthings.org on 28 August 2022. Dante at the Encyclopædia Britannica is an Italian icon.""
  },
```

Result of our query

# How about you, what do you prefer?

```
POST /my_index/_search
{
  "query": {
    "bool": {
      "must": [
        {
          "match": {
            "title": "The Godfather"
          }
        }
      ],
      "filter": [
        {
          "range": {
            "release_date": {
              "gte": "2000-01-01",
              "lte": "2022-12-31"
            }
          }
        }
      ]
    }
  },
  "highlight": {
    "pre_tags": [
      ""
    ],
    "post_tags": [
      ""
    ]
  },
  "aggregations": {
    "genres": {
      "terms": {
        "field": "genre",
        "size": 10,
        "order": "asc"
      }
    }
  }
}
```

VS



# Conclusion

**We now have access to a new range of tools**

- **Enlarge the search horizon ( multimodal, multilingual )**
- **BM25 is still in the game ( benchmark approves )**
- **Enrich your data**
- **Evaluate the impact on relevance**
- **Always track and measure relevance !**

# Some references

**Relevant Search by Doug Turnbull and John Berryman- Publisher(s): Manning Publications**

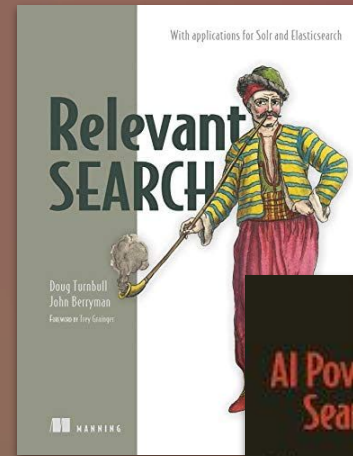
**AI powered Search by Trey Grainger, Doug Turnbull, Max Irwin - Publisher(s): Manning Publications (MEAP)**

**Adelean Blog:**

**[https://www.adelean.com/en/blog/20230401\\_nlp\\_for\\_search/](https://www.adelean.com/en/blog/20230401_nlp_for_search/)**

**Elastic Blog:**

**<https://www.elastic.co/blog/how-to-deploy-natural-language-processing-nlp-getting-started>**





adelean  
EXTRACT TRANSFORM SEARCH



[www.adelean.com](http://www.adelean.com)



[info@adelean.com](mailto:info@adelean.com)



[@a2lean](https://twitter.com/a2lean)



[linkedin.com/company/adelean](https://www.linkedin.com/company/adelean)

Thank you!

**HAYSTACK**