Max Planck Institute
for Innovation and Competition

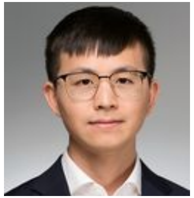
PATENTSEMTECH
2024

Logic Mill

Dietmar Harhoff - Michael Rose - Sebastian Erhardt - Mainak Ghosh - Erik Buunk - Cheng Li





SIGIR PatentSemTech Workshop - July 18, 2024 - Washington D.C.

Motivation

The **ever growing number** of **technical documents** (e.g. patents, scientific publications, standards)

How to **identify the relatedness** (proximity, similarity) of these documents?

Use metadata:

- Relations between documents (e.g., citations) – but they are **rare**
- Classifications (e.g., CPC) / Keywords / Tags - but they are **not granular** enough

Further limitations

Traditional algorithms (e.g., Bag-Of-Words, TF-IDF):

- **Ignores word order** and **context**
- **Polysemy** (multiple meanings of a word, e.g., “bat”)
- High-dimensional **sparse representations**

Off-the-shelf solutions exist, but…

- Do not allow linking **different domain corpora**
- Often use **proprietary algorithms**
- Are **not up-to-date** & **not scalable**

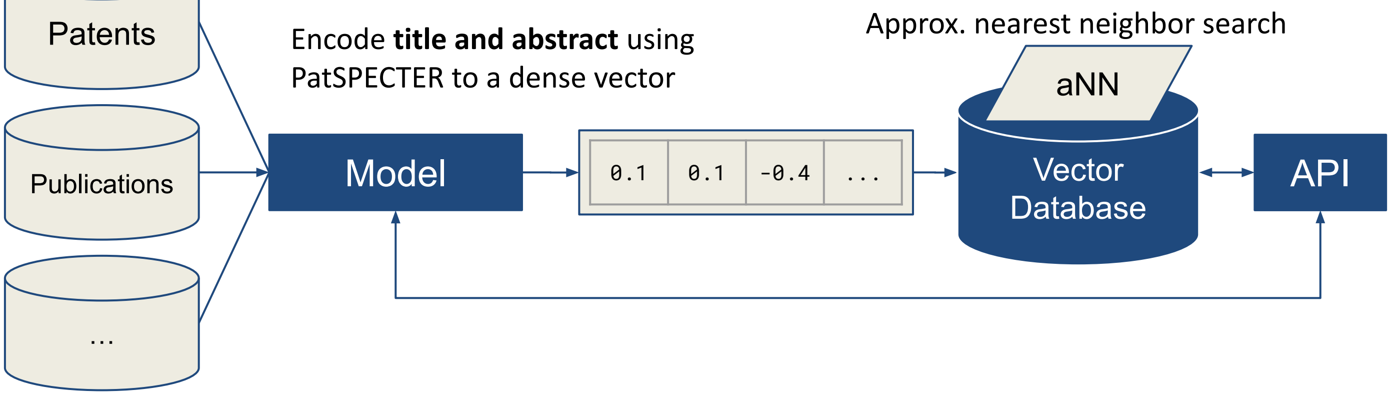
Solution

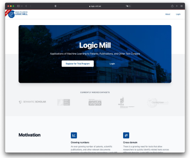
How can we overcome these shortcomings with state-of-the-art technology?

We propose a solution…

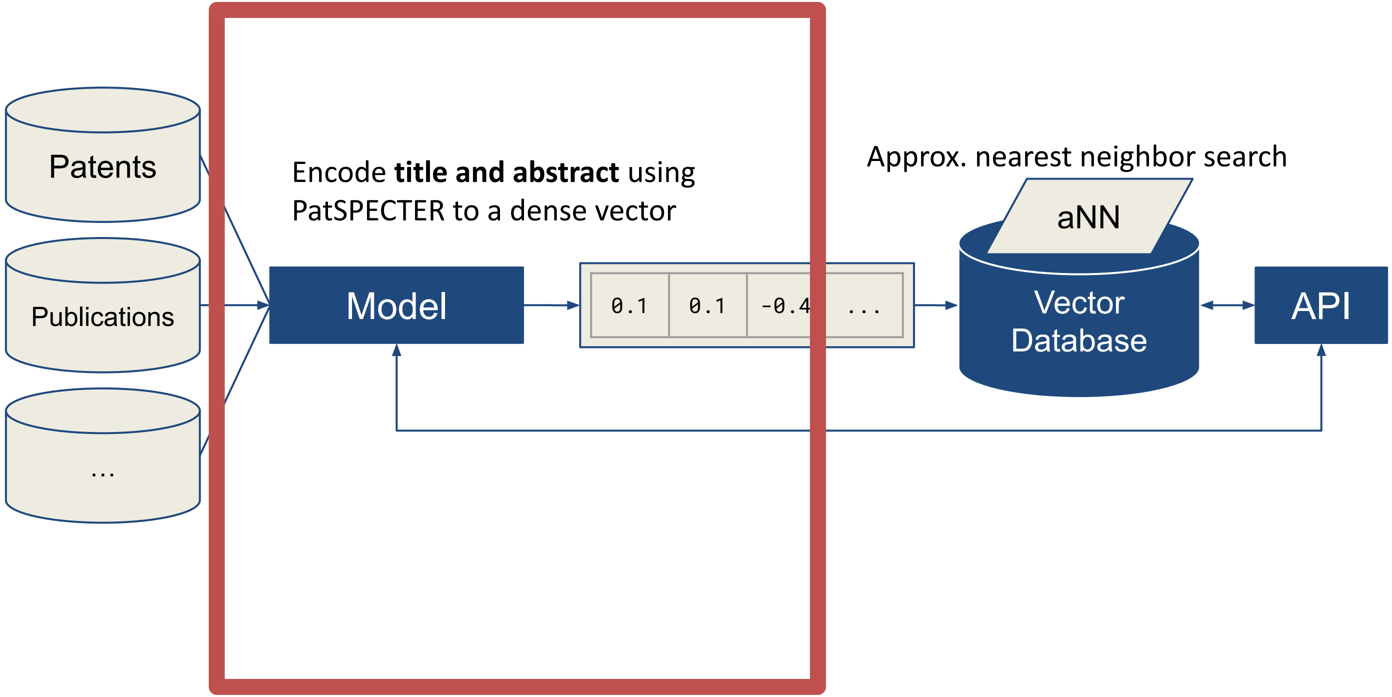
- Based on **transformer architecture language models** (BERT based)
- **Open source models**
- **Vector search database** (Elastic Search)
- **Updated continuously**
- **Accessible** via application programming interface (API)
- **Plug-and-play**

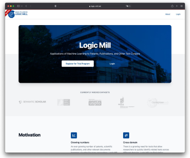
High-Level System Overview





High-Level System Overview





Model

- Initially we started our alpha version with **SPECTER (Allen AI)**
  - 768 dimensions
  - Based on **SciBERT** and trained using contrastive learning (citation information of scientific publications - Semantic Scholar)
- During the **EPO ARP** we developed 2 patent specific models
  - **PaECTER** (based on **BERT for Patents**) - 1024 Dimensions
  - **Pat-Specter** (based on **SPECTER2**) - 768 Dimensions
  - We used patent to patent citation information in conjunction with the same contrastive learning approach

Training Data - Triplets

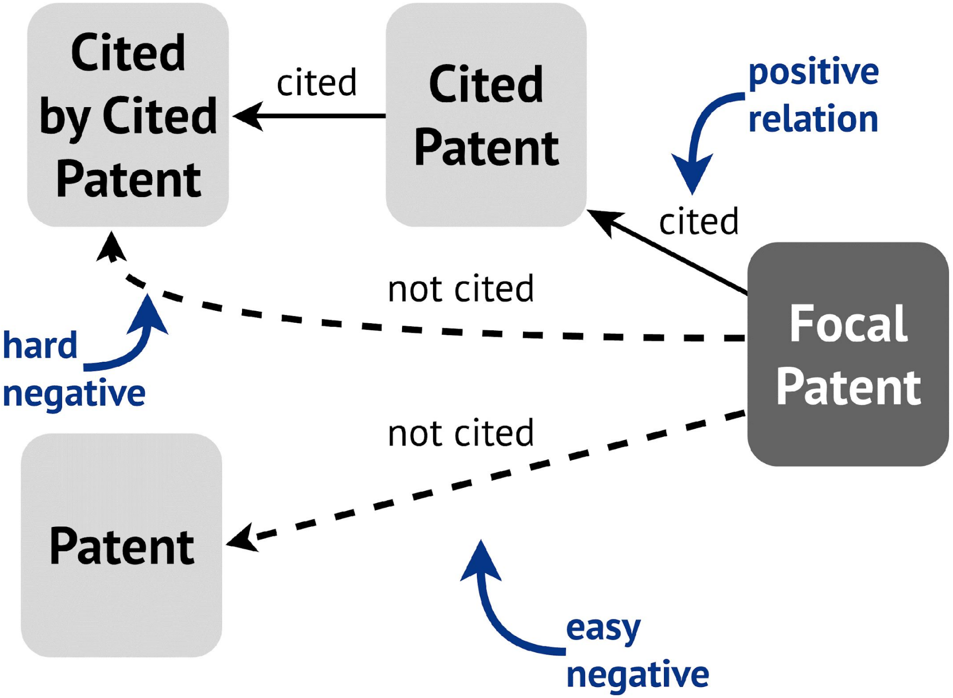
For each sampled patent, we generate 5 triplets:

```
1.   (focal patent | random* cited X/Y patent | random easy negative same CPC as focal)
2.   (focal patent | random* cited X/Y patent | random easy negative same CPC as focal)
3.   (focal patent | random* cited X/Y patent | random easy negative same CPC as focal)
4.   (focal patent | random* cited X/Y patent | random hard negative)
5.   (focal patent | random* cited A patent   | random hard negative)
```

* = drawing with replacement after available patents are used

“X” documents are documents which are highly relevant on their own.
“Y” documents deprive the claimed invention of an inventive step.
“A” documents give the general state of the art and are not considered prejudicial to the patentability of the claimed invention.

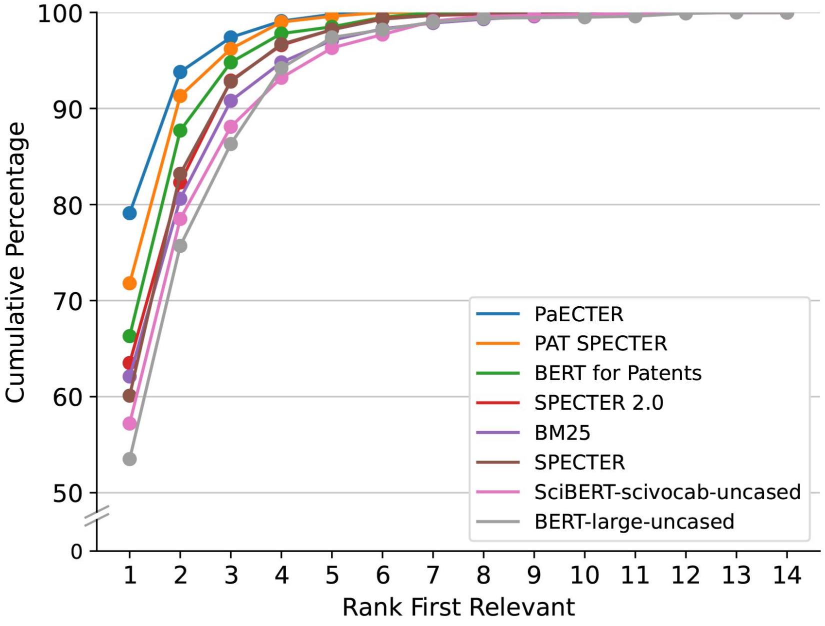
Positive and Negative Patents



Model

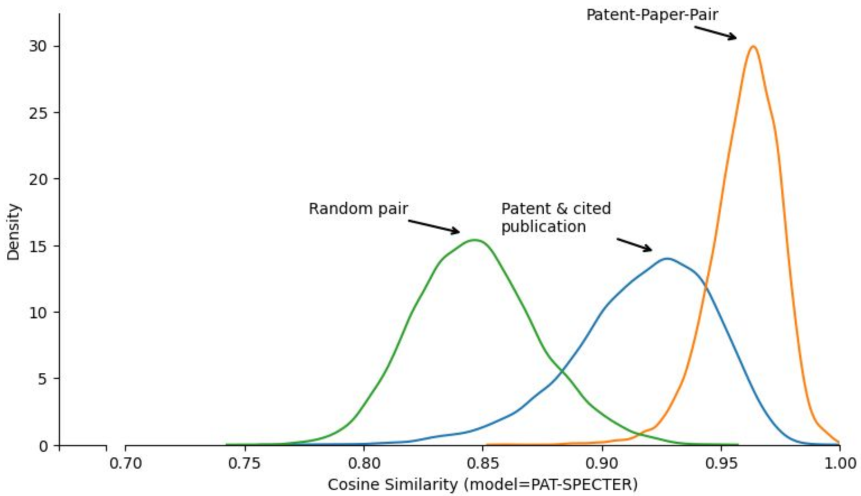
Model Comparison

Task: 5 positive and 25 random patents



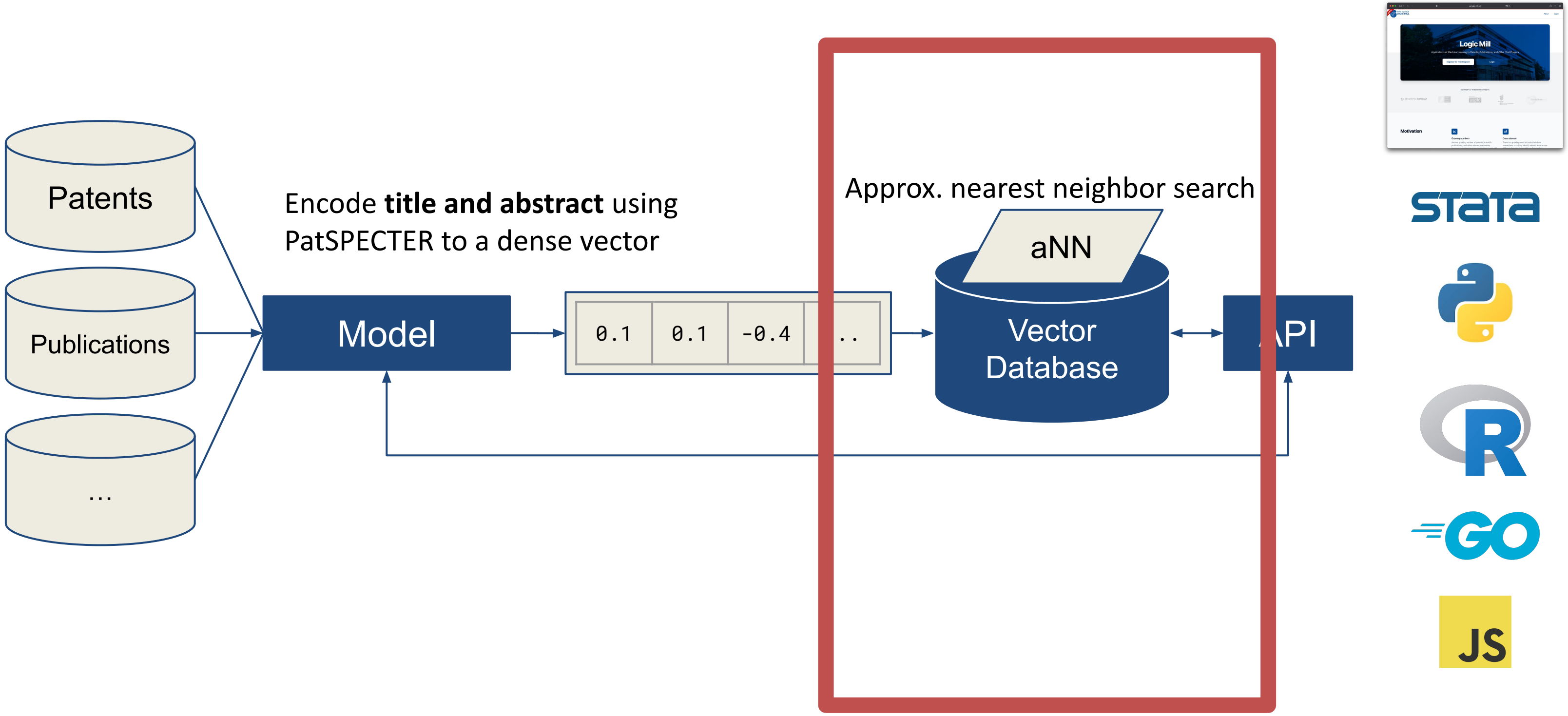
PatSPECTER

Task: Identifying Patent Paper Pairs



Also outperforms EPO's SEARCHFORMER on their own dataset (data-leakage)

High-Level System Overview



Logic Mill - Database Clusters

Cluster V1

Nodes: 12

RAM: 128GB

VCPUs: 8 VCPU

Disk: 1TB SSD

Elastic Search: 8.5.2

ANN Algo: HSNW

Data: Semantic Scholar, USPTO, EPO, WO

Cluster V2

Nodes 10

RAM: 128GB

VCPUs: 8 VCPU

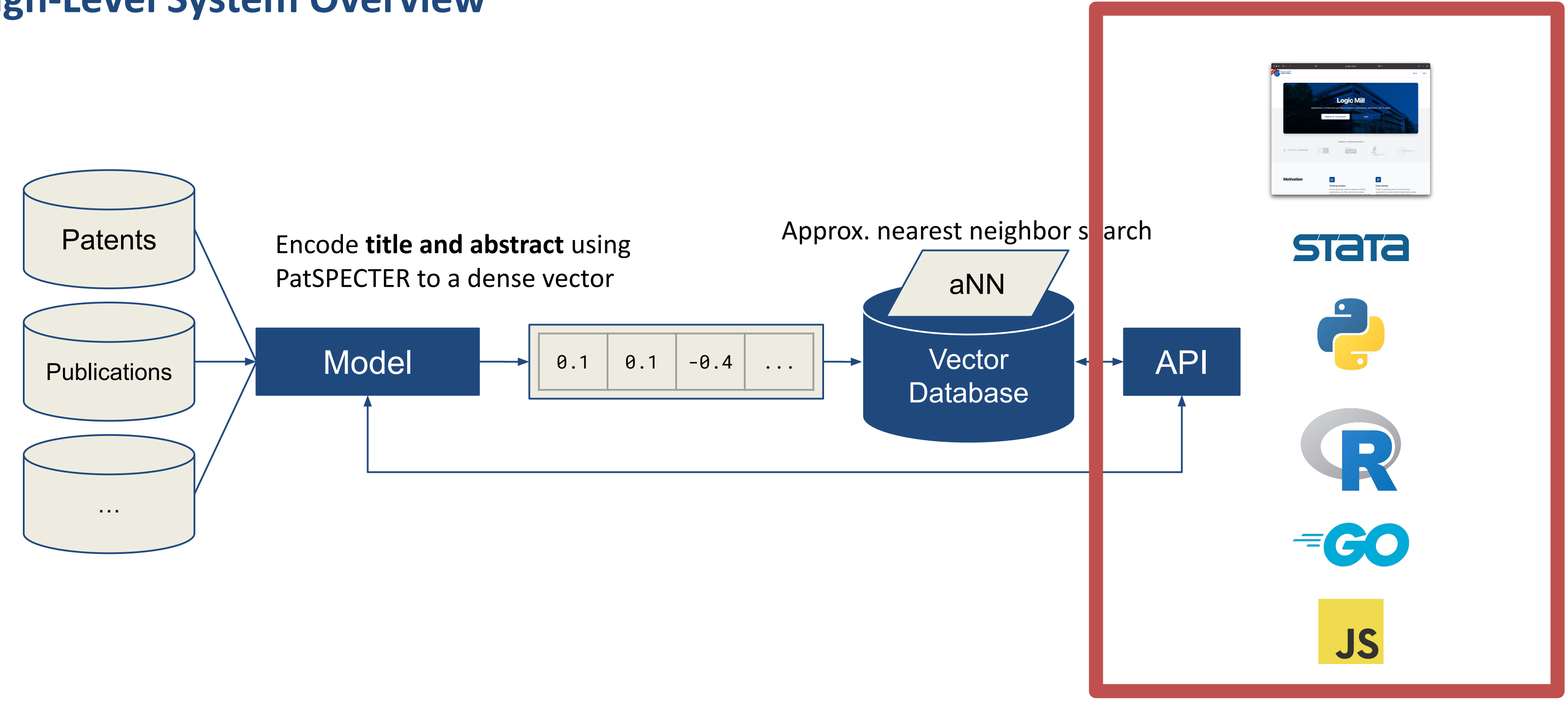
Disk: 1TB (NVME) / SSD

Elastic Search: 8.13

ANN Algo: HNSW 8bit Int

Data: Open Alex, DocDB (USPTO, EPO, WO, …)

High-Level System Overview



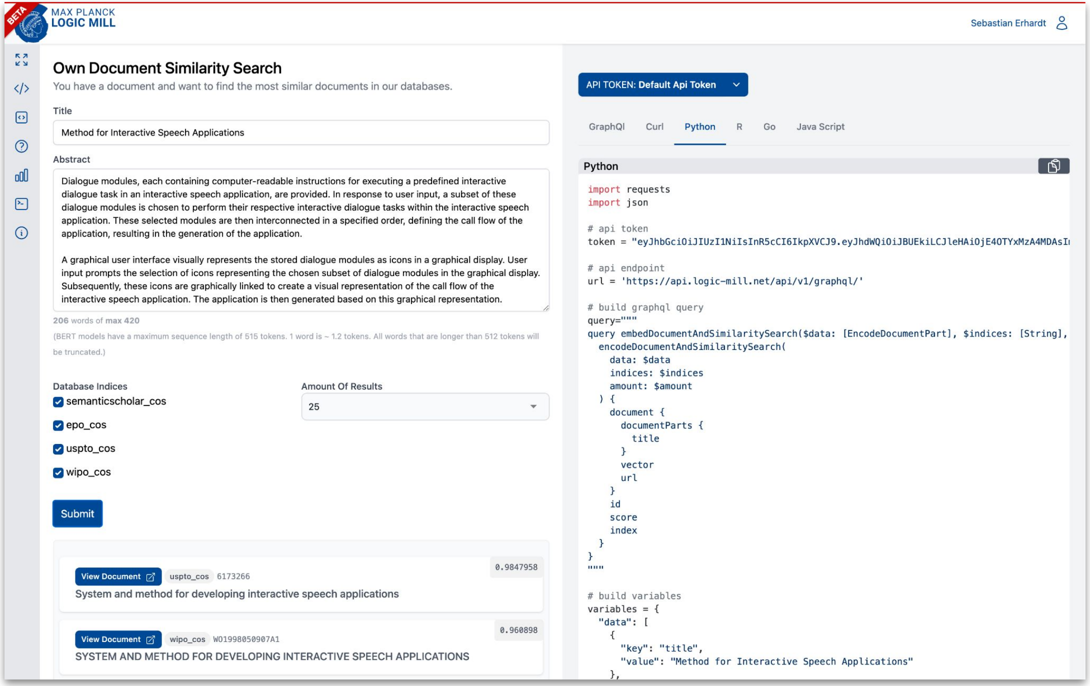
API Functionality

- Encoding (documents to vectors)
- Similarity Calculation
- Vector retrieval
- Similarity Search
  - Within the database
  - Based on a newly encoded document
- …

Demo

Title: "Method for Interactive Speech Applications"

Abstract: "Dialogue modules, each containing computer-readable instructions for executing a predefined interactive dialogue task in an interactive speech application…"



Demo

Demo.ipynb

Datei Bearbeiten Anzeige Einfügen Laufzeit Tools Hilfe Alle Änderungen wurden gespeichert

+ Code + Text

```python
        return None

    else:
        response = r.json()

        return response["data"]["encodeDocumentAndSimilaritySearch"]


# Demo

title = "Method for Interactive Speech Applications"

abstract = """
Dialogue modules, each containing computer-readable instructions for executing a predefined interactive dialogue task in an interactive speech appl

A graphical user interface visually represents the stored dialogue modules as icons in a graphical display. User input prompts the selection of ico

Additionally, the method involves associating configuration parameters with specific dialogue modules using the graphical display. Each configurati
"""

results = embedDocumentAndSimilaritySearch(title, abstract, amount=25)
for r in results:
  print(r["score"], r["index"], r["document"]["url"], r["document"]["documentParts"]["title"])
```

```
0.9847958 uspto_cos https://worldwide.espacenet.com/patent/search?q=US6173266B1 System and method for developing interactive speech applications
0.960898 wipo_cos https://worldwide.espacenet.com/patent/search?q=WO1998050907A1 SYSTEM AND METHOD FOR DEVELOPING INTERACTIVE SPEECH APPLICATIONS
0.92959857 wipo_cos https://worldwide.espacenet.com/patent/search?q=WO2006003542A1 INTERACTIVE DIALOGUE SYSTEM
0.92596334 wipo_cos https://worldwide.espacenet.com/patent/search?q=WO2020136733A1 Interactive Device, Interactive Method, And Interactive Program
0.92528945 wipo_cos https://worldwide.espacenet.com/patent/search?q=WO2007133841A2 GRAPHICAL INTERFACE FOR INTERACTIVE DIALOG
0.9180486 wipo_cos https://worldwide.espacenet.com/patent/search?q=WO2017091550A2 AUTOMATIC SPOKEN DIALOGUE SCRIPT DISCOVERY
0.9180486 wipo_cos https://worldwide.espacenet.com/patent/search?q=WO2017091550A3 AUTOMATIC SPOKEN DIALOGUE SCRIPT DISCOVERY
0.91785395 uspto_cos https://worldwide.espacenet.com/patent/search?q=US06823313B1 Methodology for developing interactive systems
0.91674423 wipo_cos https://worldwide.espacenet.com/patent/search?q=WO2000051016A1 APPARATUS FOR DESIGN AND SIMULATION OF DIALOGUE
```

Demo

Espacenet
Patentsuche

6173266

Mein Espacenet | Hilfe | Klassifikationssuche | Treffer | Erweiterte Suche | Filter | Pop-up-Tipps

Home > Treffer > US6173266B1

1. >

☆ **US6173266B1** System and method for developing interactive speech applications

Bibliografische Daten | Beschreibung | Patentansprüche | Zeichnungen | Originaldokument | Anführungen | Rechtsereignisse | Patentfamilie

Anmelder	SPEECHWORKS INT INC [US] +
Erfinder	MARX MATTHEW T [US]; CARTER JERRY K [US]; PHILLIPS MICHAEL S [US]; HOLTHOUSE MARK A [US]; SEABURY STEPHEN D [US]; ELIZONDO-CECENAS JOSE L [US]; PHANEUF BRETT D [US] +
Klassifikationen	
IPC	***G10L15/22; G10L15/26; H04M3/493; H04M3/527;*** (IPC1-7): G10L11/00;
CPC	**G10L15/22 (EP,US); H04M3/493 (EP,US); H04M3/4936 (EP,US); H04M3/527 (EP,US);** G10L2015/228 (EP); H04M2201/40 (EP,US); H04M2201/42 (EP,US); H04M2203/355 (EP,US);
Prioritäten	US4574197P·1997-05-06; US8171998A·1998-05-06
Anmeldung	US8171998A·1998-05-06
Veröffentlichung	US6173266B1·2001-01-09
Veröffentlicht als	**AU7374798A; AU758006B2; CA2292959A1; CN1163869C; CN1273661A; EP1021804A1; EP1021804A4;** US6173266B1;**WO9850907A1**

Performance

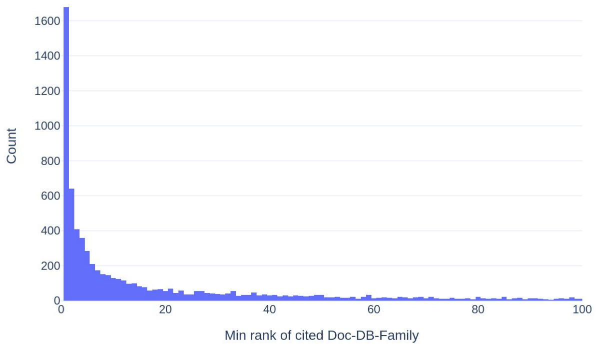
- Real World Scenario:
  Prior art search for a new patent application

- Experimental Setup:
  - We pretend to receive 2x **10,000 random patent application**
  - We take only take the **title + abstract**
  - We encode title + abstract **using PatSpecter**
  - We retrieve the **top 100 closest prior art** (patents 2 patents, patents 2 publications[1]) from our database (**published before the filing date**) only using the embedding
  - We the compare the results using ranking metrics

1) https://relianceonscience.org/patent-to-paper-citations

Patents 2 Patents

Distribution of the min Doc-DB-Family rank N=6912 (Matched)



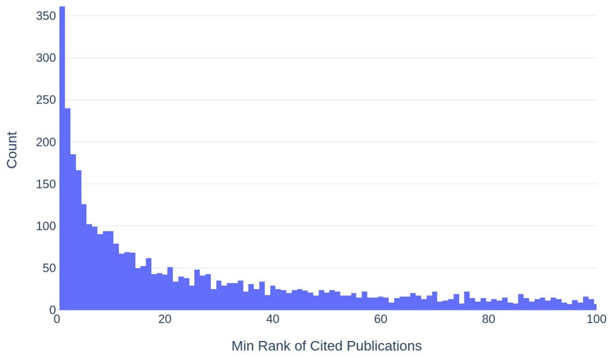
in **69%** at least 1 result in top 100

In 86% at least 1 result in top 1000

k	MRR	MAP
5	0.263874	0.062466
10	0.276368	0.065272
20	0.283207	0.070677
50	0.287384	0.078185
100	0.288638	0.082916

Patents 2 Publications

Distribution of the Min Rank N=3740 (Matched)



in **37%** at least 1 result in top 100

k	MRR	MAP
5	0.162932	0.052162
10	0.179573	0.066367
20	0.190259	0.080288
50	0.197886	0.093623
100	0.200598	0.100609

Potential Applications

- Prior art search
- Similarity document analysis & recommendation
- Patent novelty analysis
- Clustering
- Tracing of knowledge flows
- Trend analysis
- Patent portfolio analysis
- **Patent landscaping (Next Presentation)**
- …

Current Status

- Transition phase
- **8** different API **functions**
- Tutorials / sample codes
- **228M+** Documents:

  7M+ EPO → DocDB EPO

  13M+ USPTO → DocDB USPTO

  3M+ WIPO → DocDB WIPO

  205M+ Semantic Scholar → Open Alex
- **190+** Users from **40+** Institutions
- **12M+** API requests

Road Map

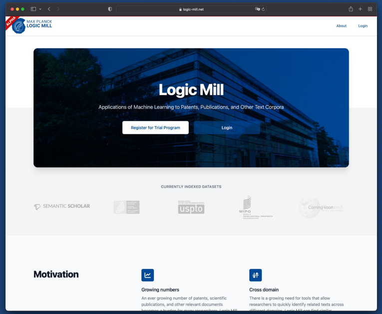
- Next-Generation Model:

  → Overcome the 512 **token limit**
- Extend API **functionality**
- Provide **pre-computed datasets**

  E.g. **distances from EP patents to EP patents**
- …

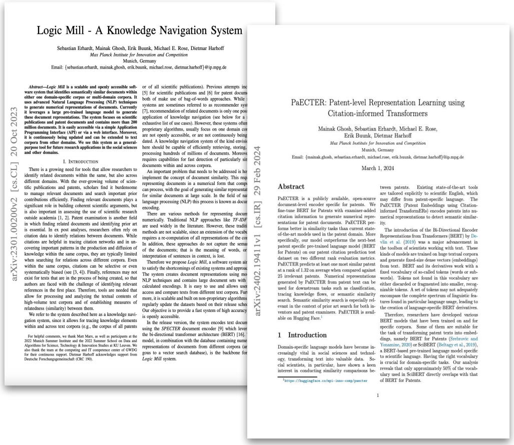
How to get started?

- Apply via https://logic-mill.net/
- Check out documentation
- Use Python, R, Stata, … to pull data through our API (the website generates the code for you)

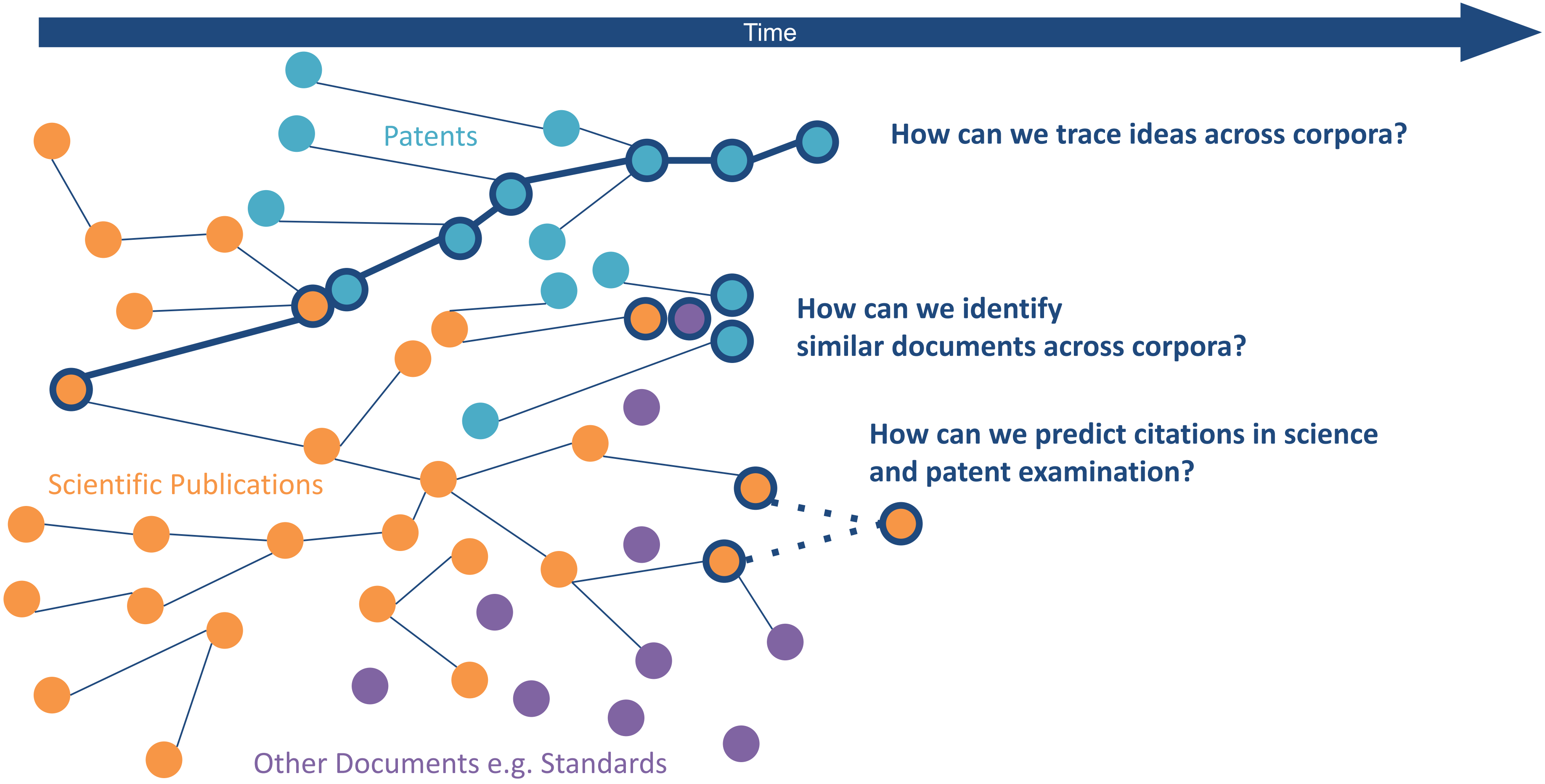


Cite our papers

- https://arxiv.org/abs/2301.00200
  Logic Mill - A Knowledge Navigation System
- https://arxiv.org/abs/2402.19411
  PaECTER: Patent-level Representation Learning using Citation-informed Transformers



Appendix


Time
Patents
How can we trace ideas across corpora?
How can we identify
similar documents across corpora?
How can we predict citations in science
and patent examination?
Scientific Publications
Other Documents e.g. Standards

Overview 2024

OpenAlex

UNITED STATES PATENT AND TRADEMARK OFFICE uspto

WIPO WORLD INTELLECTUAL PROPERTY ORGANIZATION

Europäisches Patentamt
European Patent Office
Office européen des brevets

EPO DOC DB

WIKIPEDIA The Free Encyclopedia

soon

other

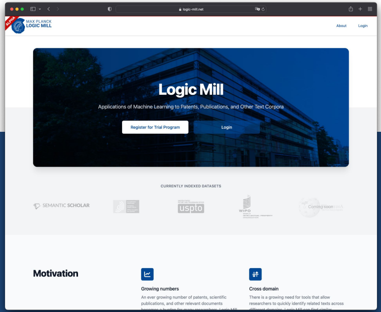
MAX PLANCK
LOGIC MILL
BETA VERSION

Document Encoding **Document Similarity** **Similar Document Search**

API

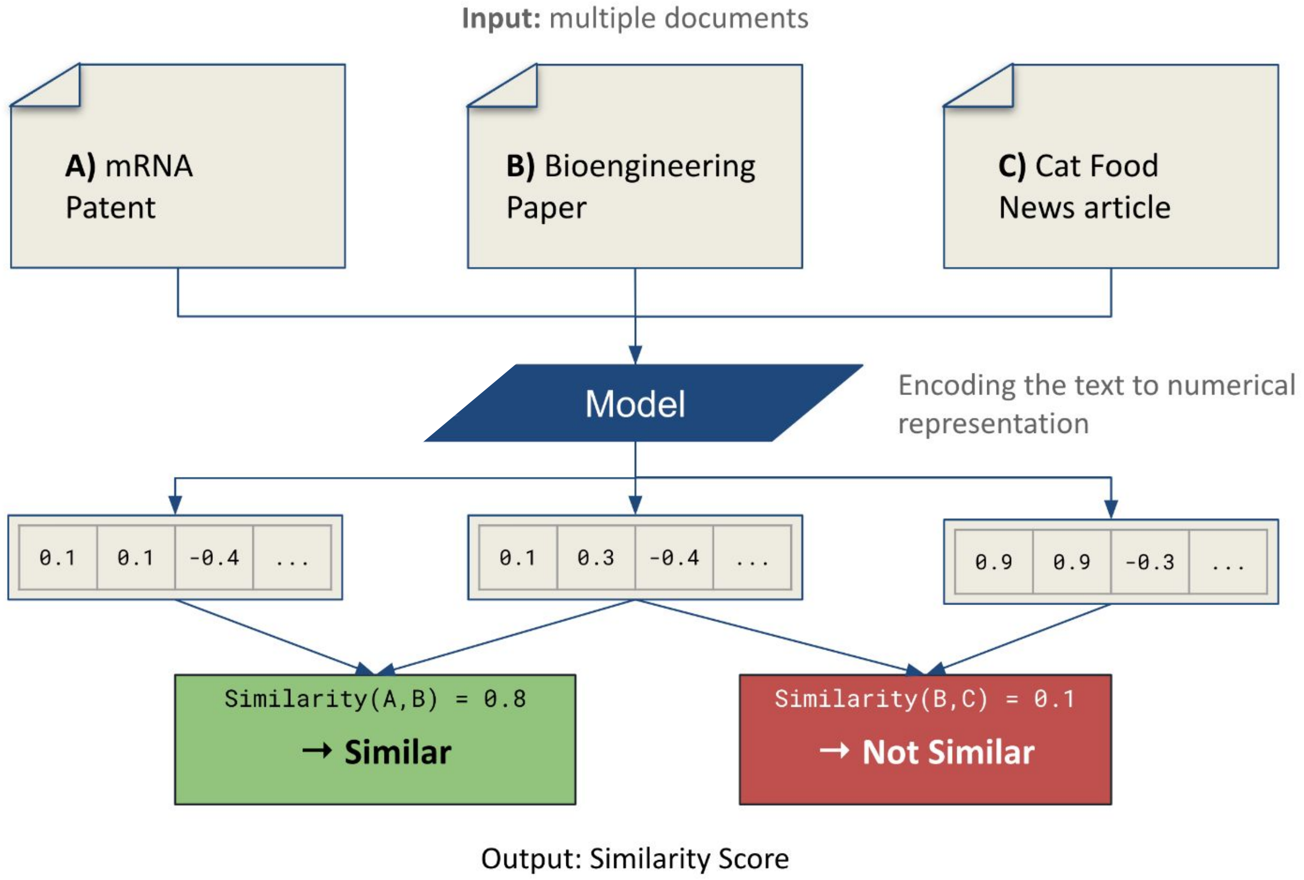
STATA

JS



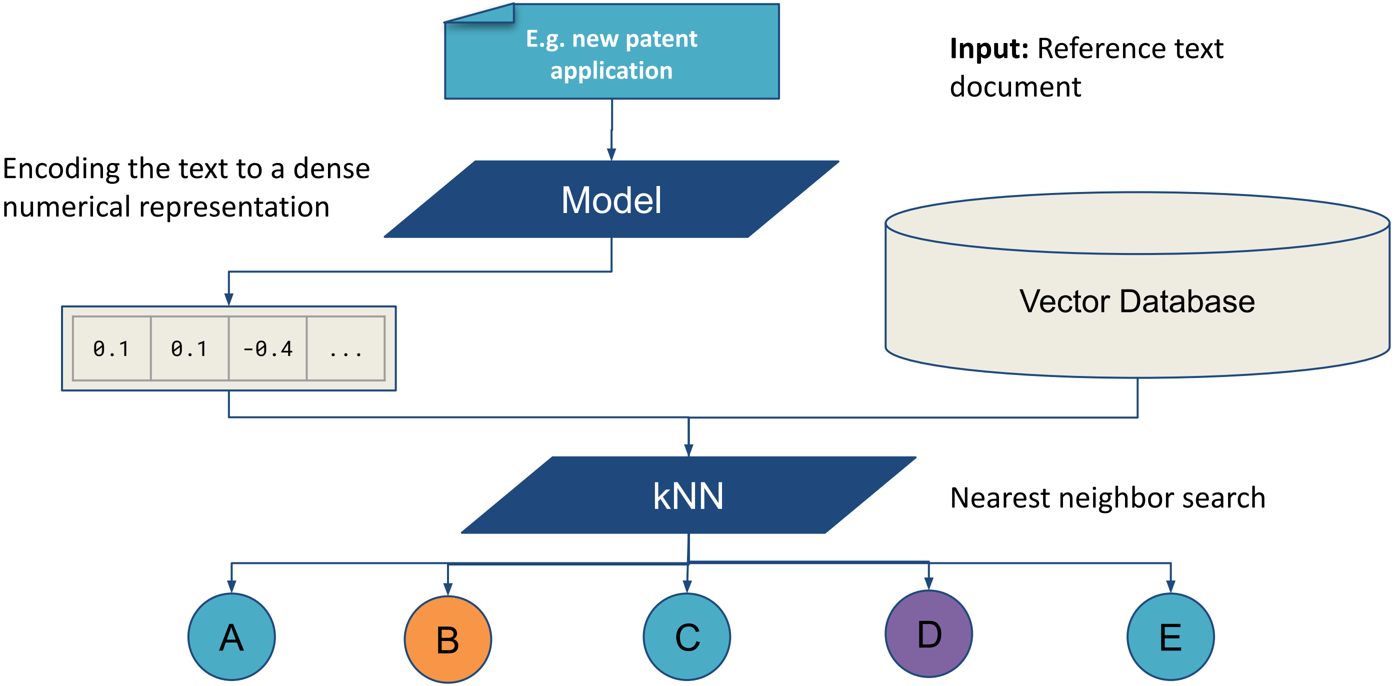
Info & registration: logic-mill.net

Document Similarity

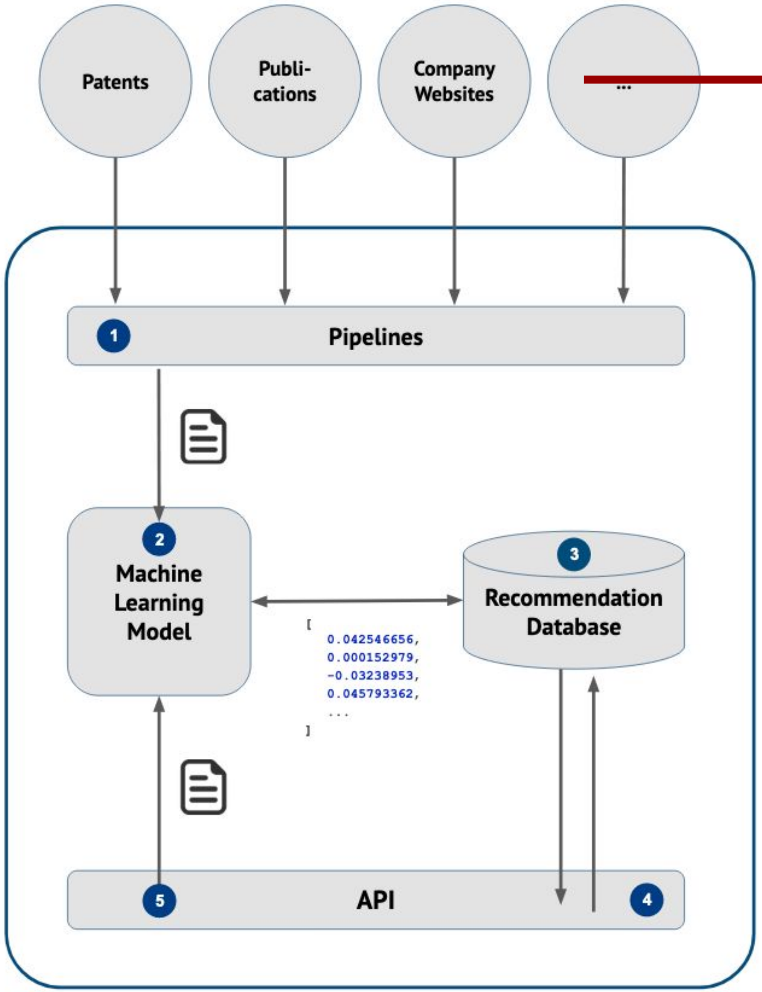


MAX PLANCK LOGIC MILL
ALPHA VERSION

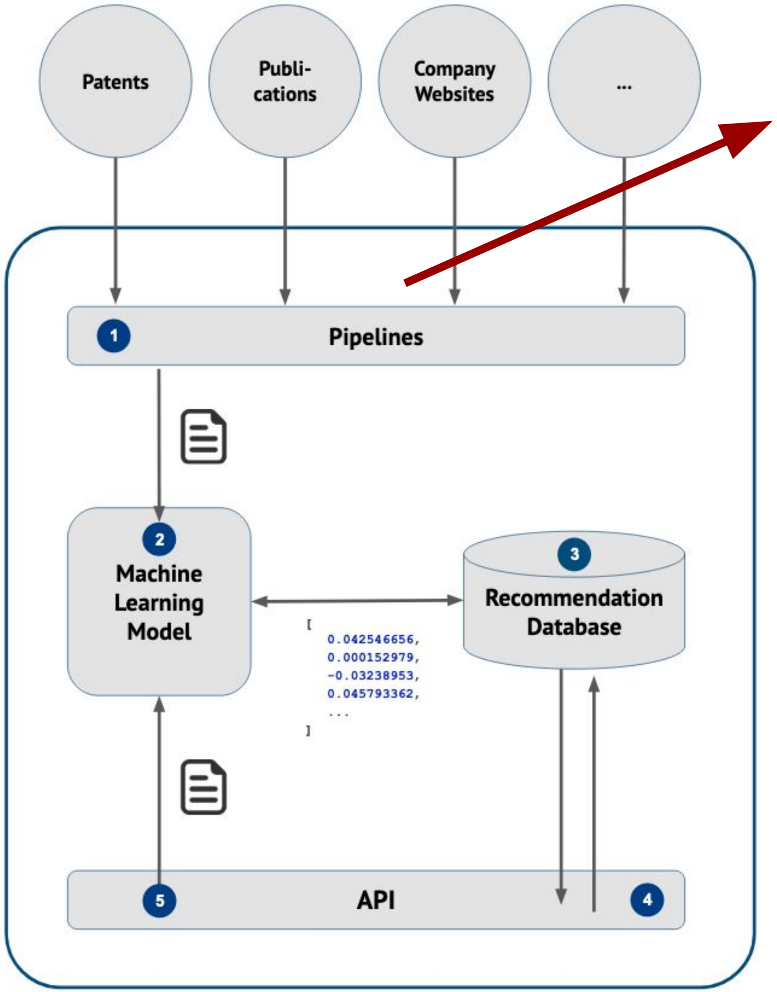
Vector Database



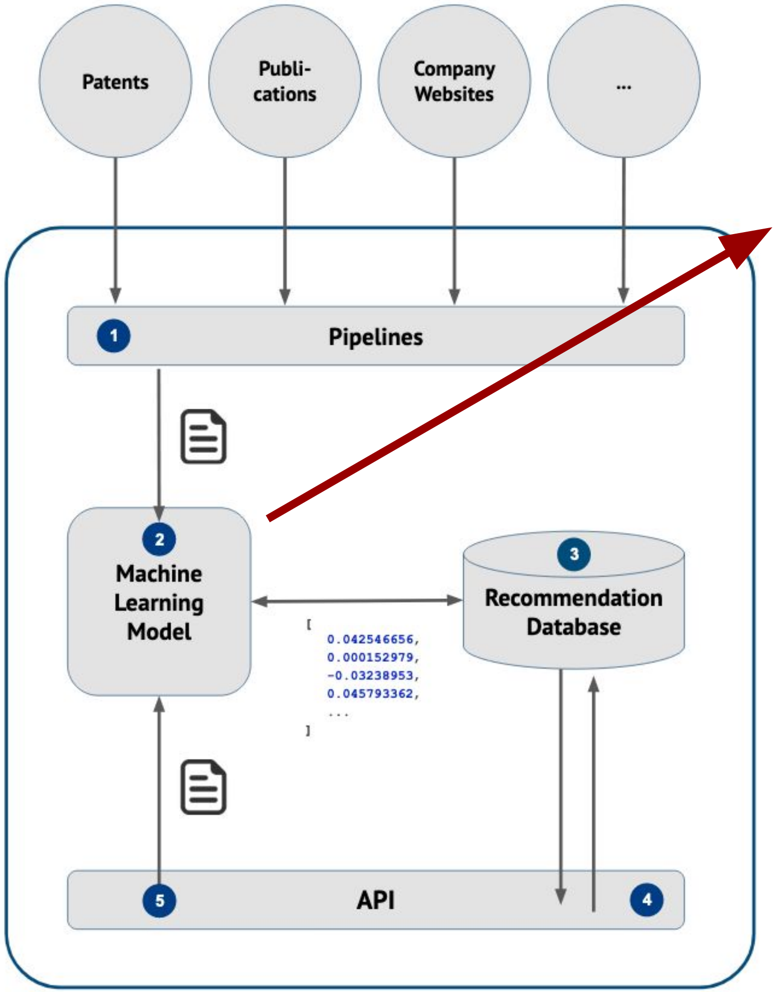
Output: Nearest neighbor documents of the reference document
e.g. patents, scientific publications, standards …



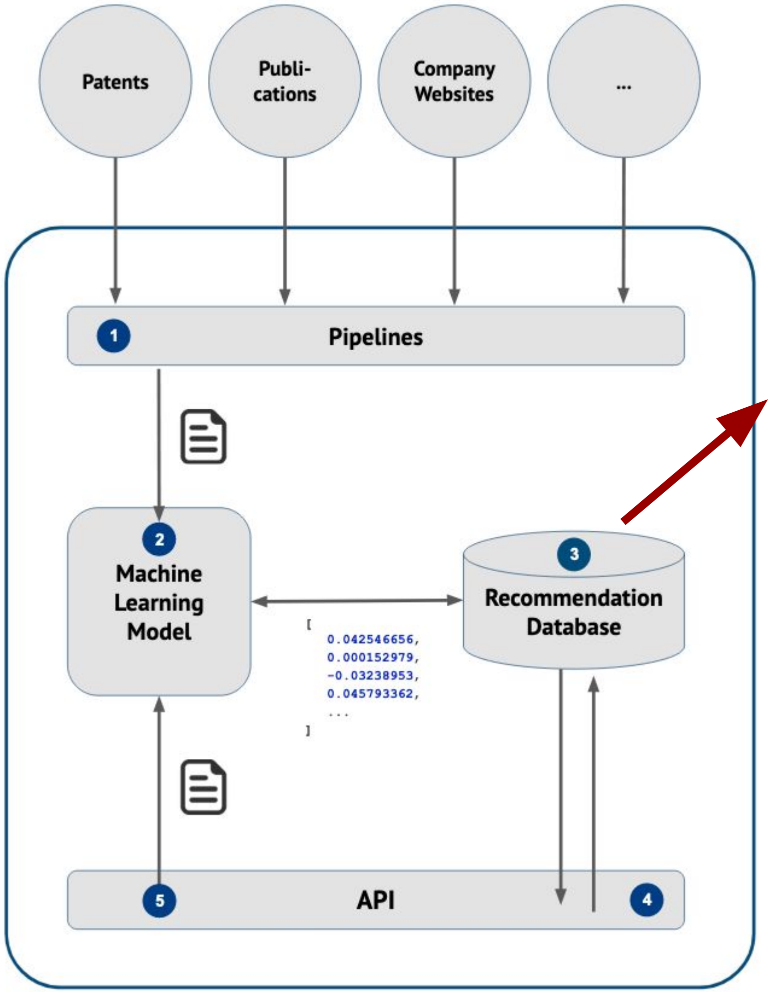
Multiple corpora....



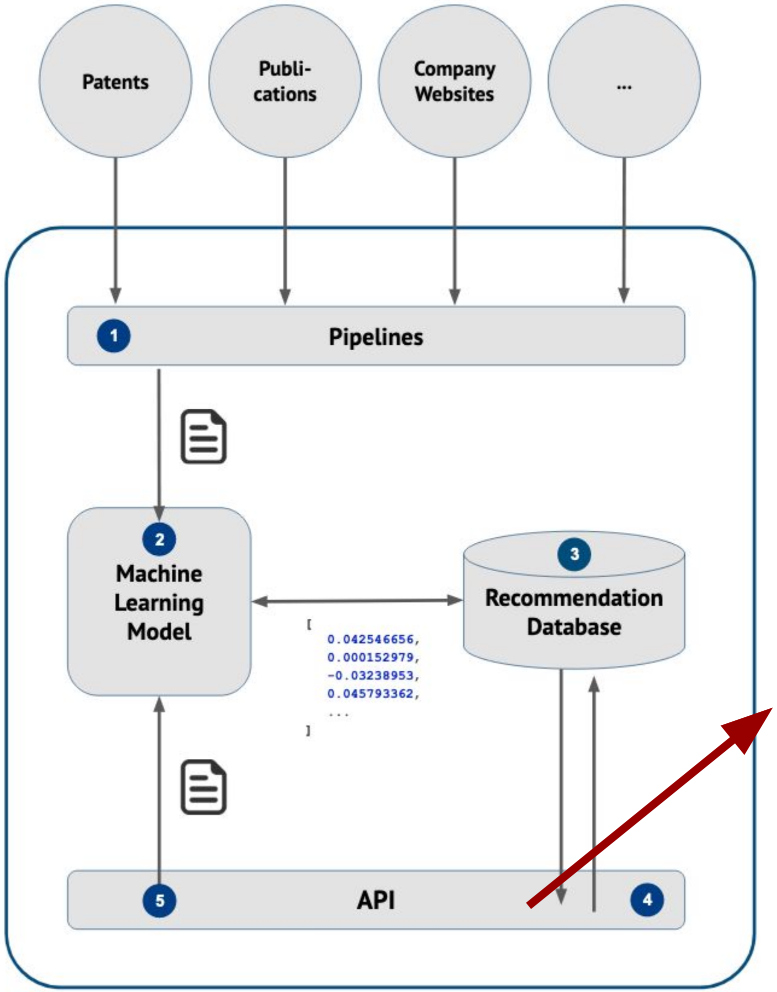
1 Ingest data using pipelines for **automation and scalability**



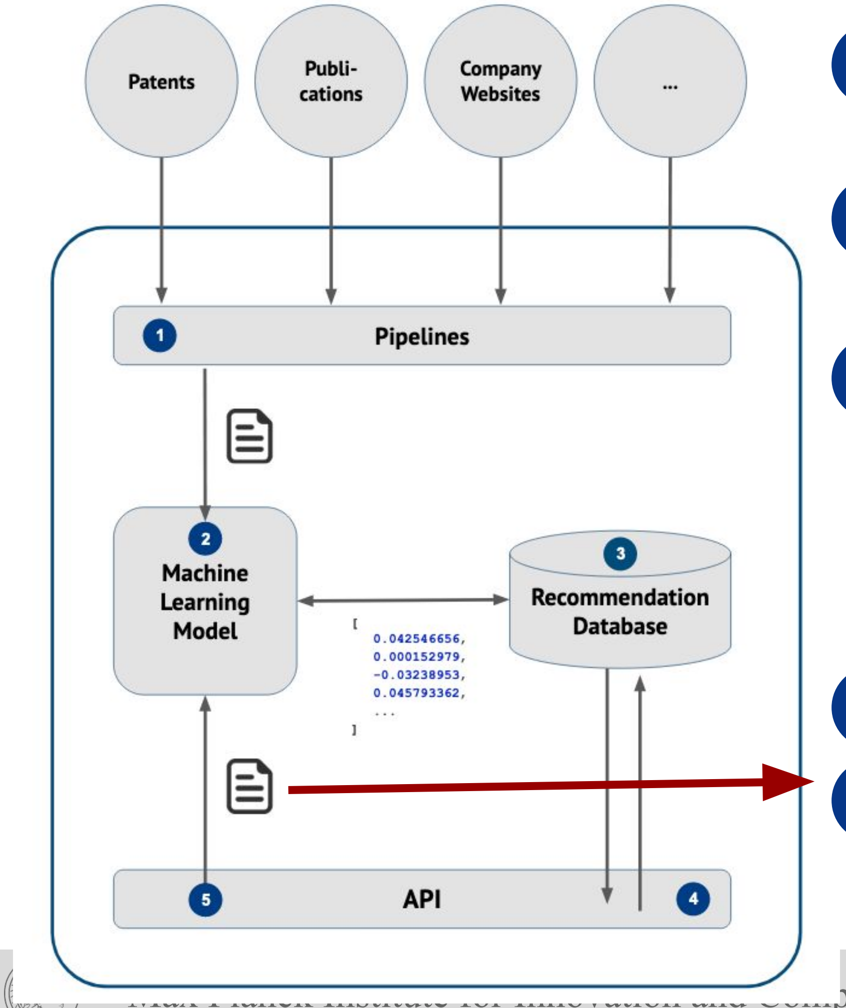
1. Ingest data using pipelines for **automation and scalability**
2. Encode title and abstract using SPECTER to a numerical representation



1. Ingest data using pipelines for **automation and scalability**
2. Encode documents with a **deep learning** model to a numerical representation
3. Store the representation in a database and recommend relevant documents using **similarity operators** on the numerical representation



1. Ingest data using pipelines for **automation and scalability**
2. Encode documents with a **deep learning** model to a numerical representation
3. Store the representation in a database and recommend relevant documents using **similarity operators** on the numerical representation
4. User interaction via public **API**

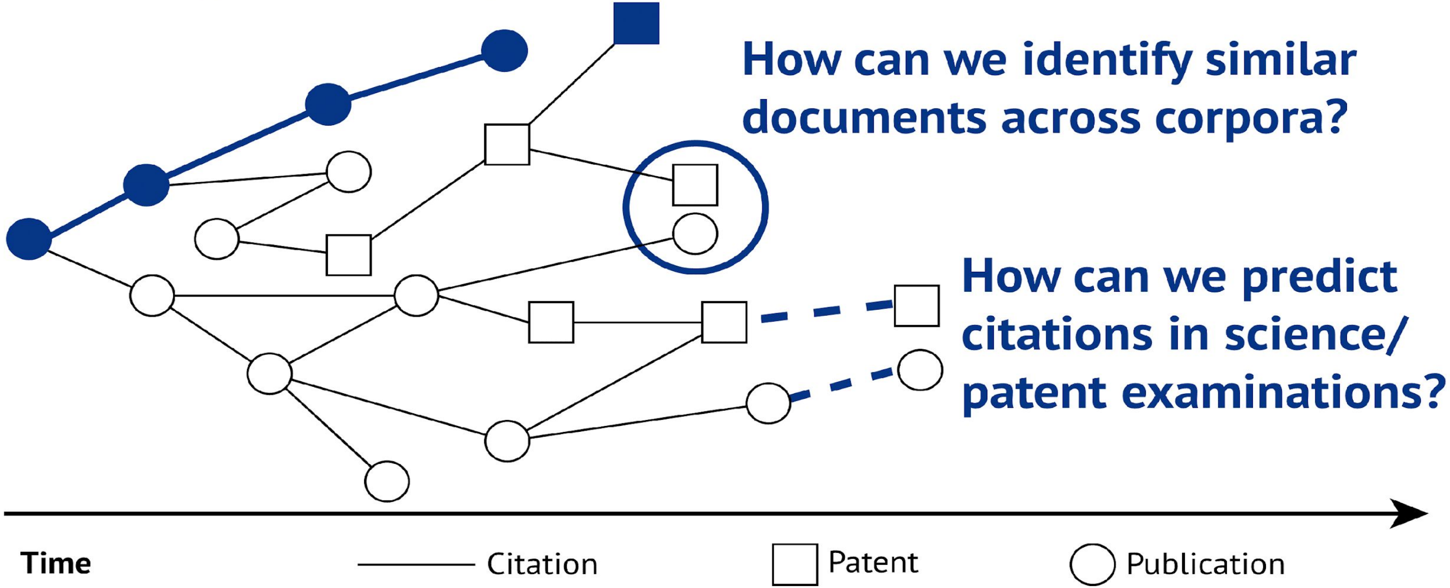


1. Ingest data using pipelines for **automation and scalability**
2. Encode documents with a **deep learning** model to a numerical representation
3. Store the representation in a database and recommend relevant documents using **similarity operators** on the numerical representation
4. User interaction via public **API**
5. Options for uploading **user owned text** corpora

How can we trace the flow of knowledge across corpora?

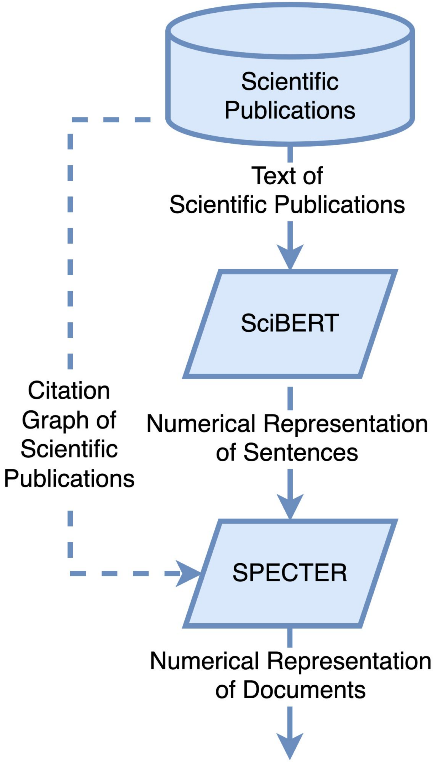
How can we identify similar documents across corpora?

How can we predict citations in science/patent examinations?



SPECTER

Scientific **P**aper **E**mbeddings using **C**itation-informed **T**ransform**ER**s

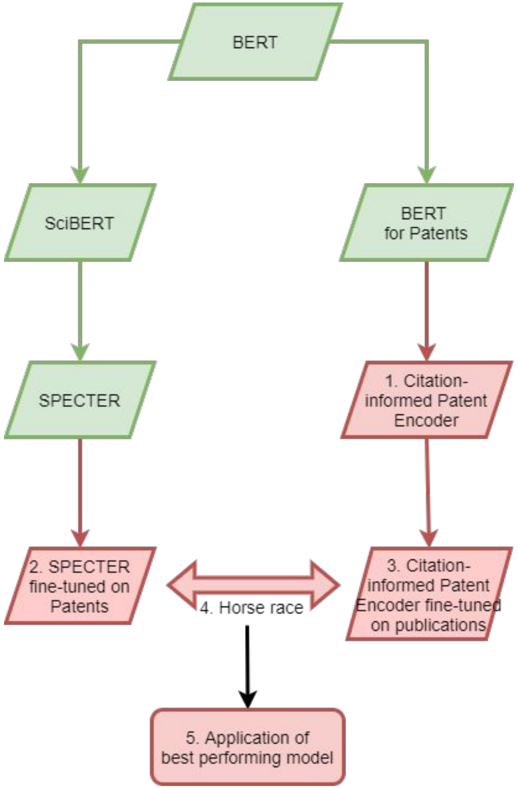


- SciBERT - A deep learning model that was pre-trained on **scientific publications** (Beltagy et al 2019)

- SPECTER - Uses the numerical representations SciBERT generates and plugs in the **citation knowledge** to derive a notion of similarity / dissimilarity between documents (Cohan et al 2020)

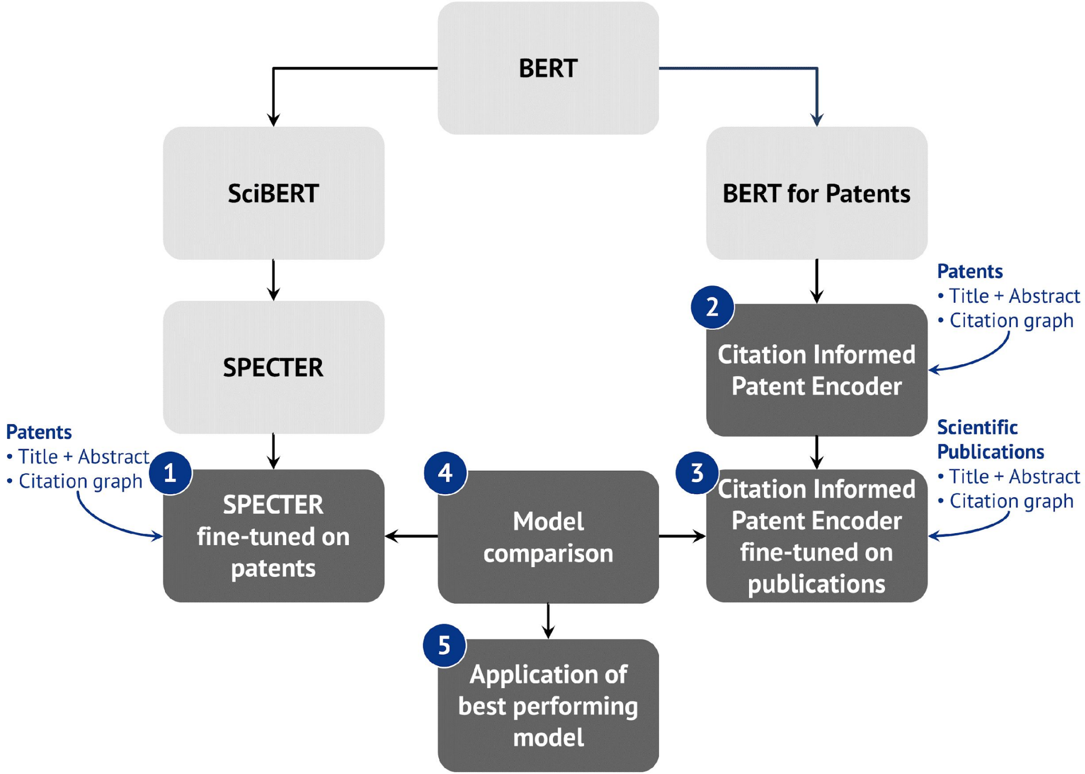
- Patent data - the model showed promising results, but due to a clear **difference** in the **vocabulary**, in the **document structure** and the **citation graphs**, an improved model is needed

EPO ARP Grant - Explore Model

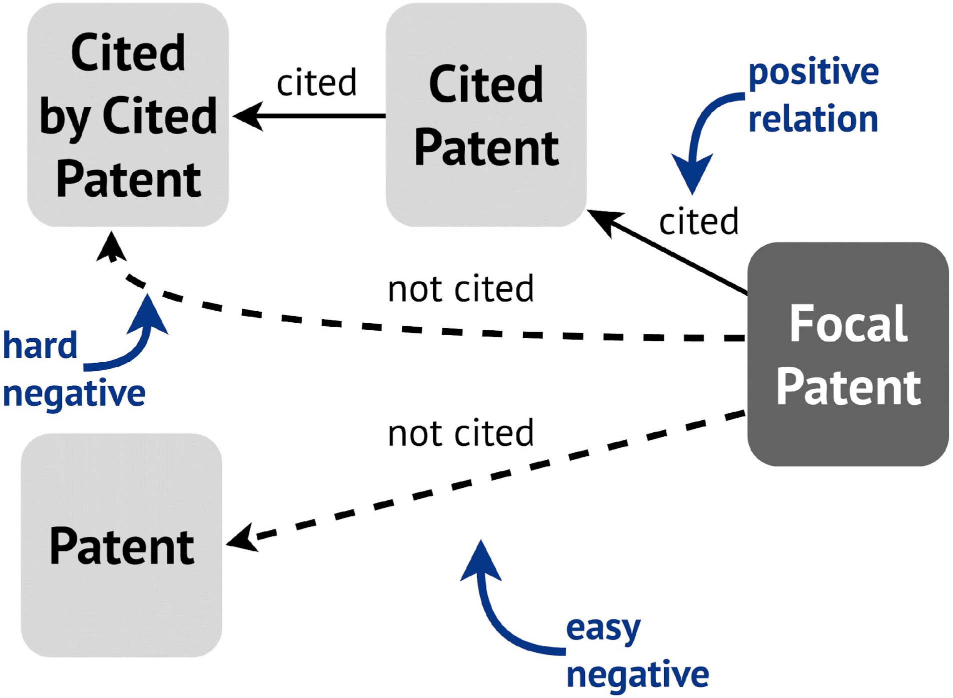


1. Build a Citation Informed Patent Encoder based on Google's *BERT for Patents* → Working Title: PaECTER
2. Fine-tune the existing SPECTER model with patents and the patent citation graph
3. Fine-tune the PaECTER with scientific articles and the science citation graph
4. Compare the performance of the models
5. Apply the best model in a real-world scenario

Project overview



Positive and negative patents



Triplets

For each sampled patent, generate 5 triplets:

```
1.   (focal patent | random* cited X/Y patent | random easy negative same CPC as focal)
2.   (focal patent | random* cited X/Y patent | random easy negative same CPC as focal)
3.   (focal patent | random* cited X/Y patent | random easy negative same CPC as focal)
4.   (focal patent | random* cited X/Y patent | random hard negative)
5.   (focal patent | random* cited A patent   | random hard negative)
```

* = drawing with replacement after available patents are used

English substitutes

- We train on title and abstract
- Title and abstract must be in English
- We substitute with the best possible English abstract if abstract is unavailable from the same DOCDB family
- We use the following order in the selection:

WO > US > GB > CA > AU > DE > CN > TW > KR > FR > JP

Triplet Margin Loss

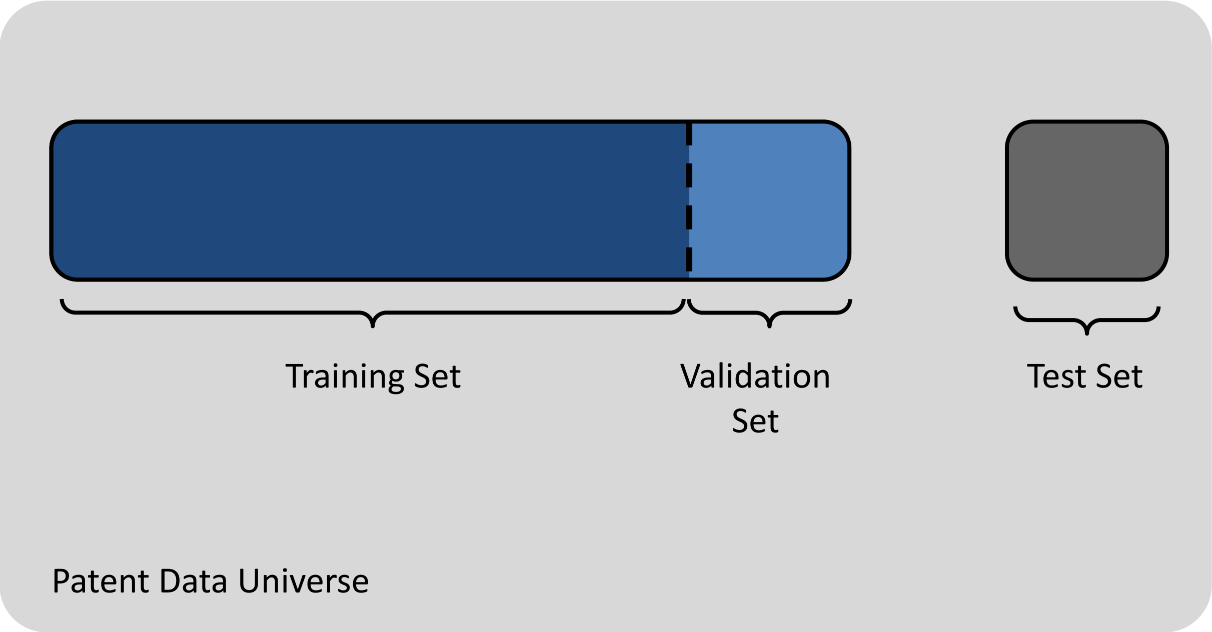
Minimize the **triplet margin loss** function during training:

$$max\{(\|V_F - V_P\|_2 - \|V_F - V_N\|_2 + m), 0\}$$

Where

- $\|.\|_2$: L2 norm distance
- V_F , V_P , V_N: numerical representations for focal patent (P_F), positive patent (P_P), and negative patent (P_N)
- *m*: The corresponding positive patent is at least *m* distance units closer to its focal patent than the corresponding negative patent (margin)

Training, Validation, Test datasets



Training/Validation (80/20 split) Dataset:

150k triplets:

- 1 focal
- 1 positive
- 1 negative

Test Dataset:

1k triplets:

- 1 focal
- 5x positive
- 25x negative

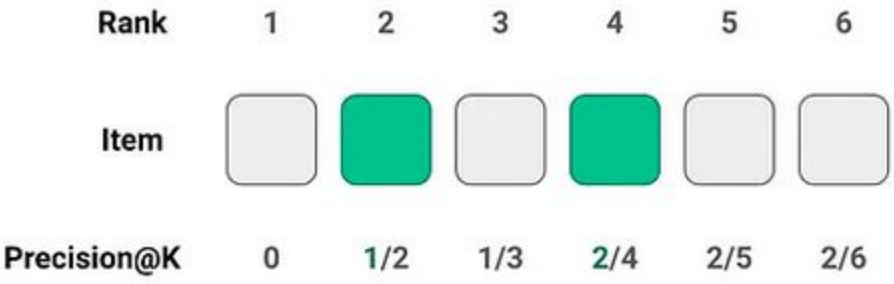
Training of the models

- **Evaluation** during training every 2000 triplets and save the best model
- **Hyper parameters** to tune:
  – Learning rate
  – Margin
  – Batch size
  – Number of epochs
  – Number of GPUs and number of nodes
  – Train/validation split

Benchmark against earlier models using Rank-aware evaluation

Model	MAP	MRR@10
SPECTER	55.87	76.83
SPECTER 2	56.79	79.25
PAT SPECTER (SPECTER fine-tuned on patents)	62.76	82.42
BERT for Patents	59.75	80.16
PaECTER (BERT for Patents fine-tuned on patents)	**68.18**	**86.66**

Average Precision at k (AP@k) *[AveP in next slide]*



$$\mathrm{AP@6} = \frac{1}{2}(0 \cdot 0 + 0.5 \cdot 1 + 0.33 \cdot 0 + 0.5 \cdot 1 + 0.4 \cdot 0 + 0.33 \cdot 0)$$

$$\mathrm{AP@6} = 0.5$$

Relevant item Irrelevant item

https://towardsdatascience.com/mean-average-precision-at-k-map-k-clearly-explained-538d8e032d2

Mean Average Precision at k (MAP@k)

Average Precision (AP or AveP) is calculated by considering the precision at each position in the ranked list where a relevant item is found, and then averaging these precisions.

$$\mathrm{AveP} = \frac{\sum_{k=1}^{n} P(k) \times \mathrm{rel}(k)}{\text{total number of relevant documents}}$$

where rel(*k*) is an indicator function equaling 1 if the item at rank k is a relevant document, 0 otherwise

MAP is obtained by averaging the AveP values over multiple queries.

$$\mathrm{MAP} = \frac{\sum_{q=1}^{Q} \mathrm{AveP(q)}}{Q}$$

where Q is the number of queries.

https://en.wikipedia.org/wiki/Evaluation_measures_(information_retrieval)#Mean_average_precision

Mean Reciprocal Rank (MRR)

$$\text{MRR} = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{\text{rank}_i}.$$

where rank_i refers to the rank position of the *first* relevant document for the *i*-th query.

Query	Proposed Results	Correct response	Rank	Reciprocal rank
cat	catten, cati, **cats**	cats	3	1/3
torus	torii, **tori**, toruses	tori	2	1/2
virus	**viruses**, virii, viri	viruses	1	1

Given those three samples, we could calculate the mean reciprocal rank as (1/3 + 1/2 + 1)/3 = 11/18 or about 0.61.

https://en.wikipedia.org/wiki/Mean_reciprocal_rank