Biomedical Text Mining
Hands-on Session

AI4Health Winter School
7-8th January 2021

ai4healthschool.org

Chryssa Zerva
Fenia Christopoulou

NaCTeM

Research Staff










Visitors

Students







Current Research Topics

- Named Entity Recognition
  - Entity Normalisation
  - Coreference Resolution
- Relation Extraction
  - Supervised Sentence/Document-level
  - Detection of Temporal links
  - Unsupervised/Open
- Event Extraction
  - Nested Event Extraction
  - Uncertainty Identification
  - Document level EE
- Cancer omics text mining
- Screening for systematic reviews
- Scientific citation extraction
  - Scientific summarisation
- Misleading information detection
  - Corroborating/contradicting evidence
  - Rank by trust/confidence
- Development of annotated corpora
  - Active/Pro-active learning
  - Crowdsourcing
- Emotion Detection
  - Social media for ADR
- Risk Assessment

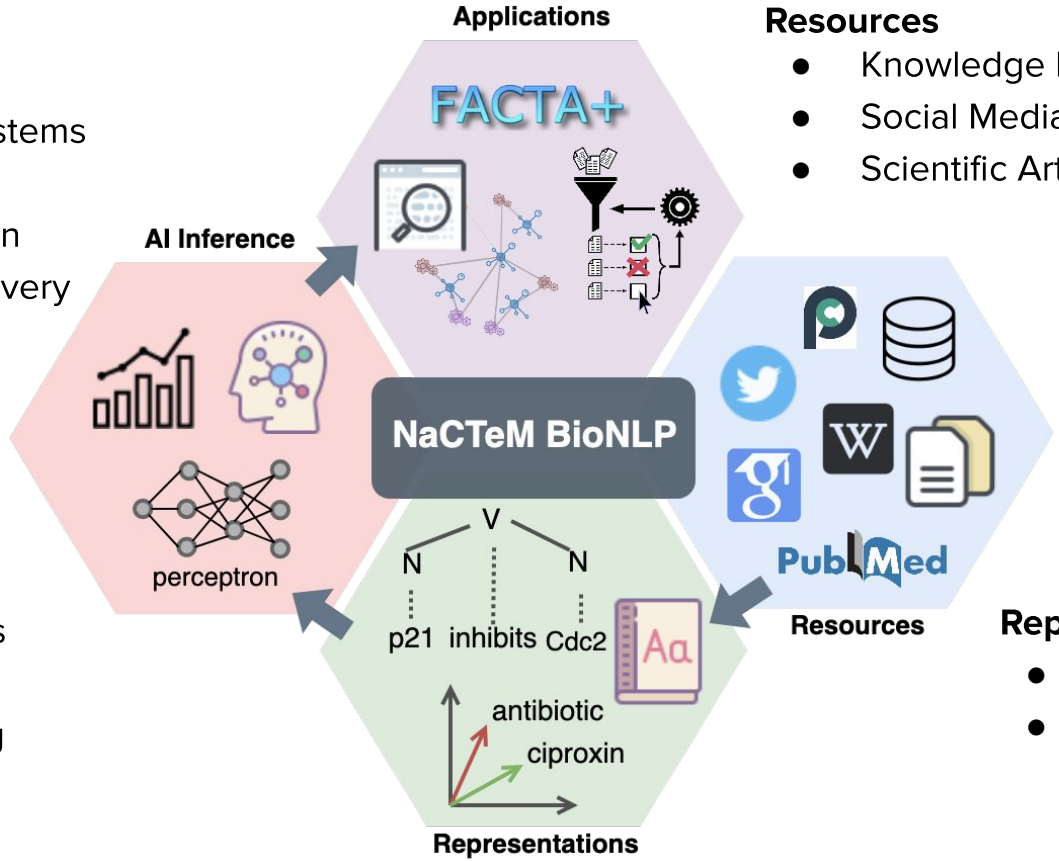
Overview

1st part	Introduction to BioNLP Tasks and related applications	Application demos and try-out
2nd part	Theoretical background: - language representations - model architectures	
3rd part	Hands-on session: NE, RE & EE on biomedical data	Move to colab!
4th part	Hands-on session: NE, RE & EE on biomedical data	Move to colab!

Introduction

Applications
- Search-based Systems
- Pathway Models
- Automatic curation
- Knowledge Discovery

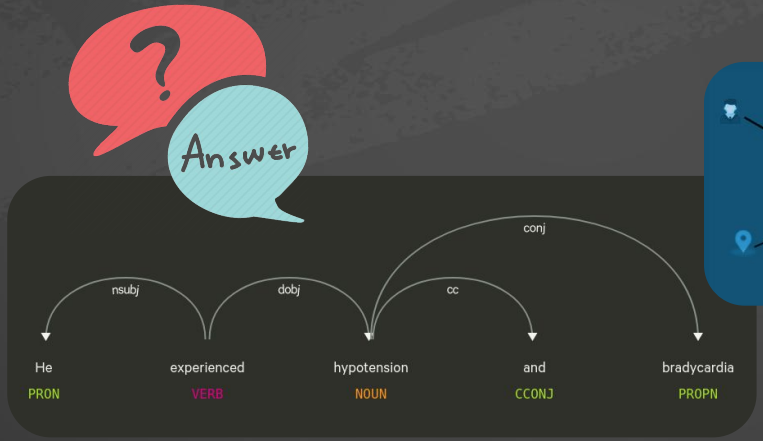
Inference
- Statistic Models
- Language Models
- NLP
- Machine Learning



Resources
- Knowledge Bases
- Social Media
- Scientific Articles

Representations
- Embeddings
- Language Models

BioNLP Tasks



BioNLP Tasks

Key Tasks

Covered in this session

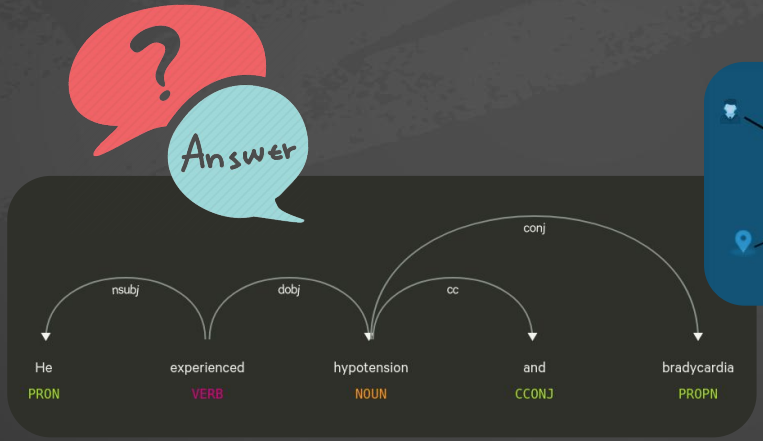
- Named Entity Recognition
- Relation Extraction
- Event Extraction

Additional Tasks

Briefly mentioned in this session

- Entity Normalisation
- Metaknowledge/Modality/Negation
- Text Summarisation
- Question Answering

key BioNLP Tasks



Named Entity Recognition

➜ The task that aims to **identify named entities** in **text**

EGTA inhibited down-regulation of alpha-AR mRNA by NE.

Chemical Protein

Input: A textual snippet (typically a sentence)
Output: A label for each token in the sentence

Definitions:

- **Named Entity**: A word or a group of words that constitute a proper name (e.g. EGTA)
- **Entity Type**: Semantic category in which a named entity belongs (e.g. Chemical)
- **Entity Span**: The words that are included in the entity (e.g. alpha-AR: 57-65)

Notes:

- Named Entity types are typically *pre-defined*
- Current systems have reached almost human-level performance on NER
  But, when using very *general,high-level categories* (check)
- *Tokenization* is essential before NER!

NER: Tagging Schemes

- NER is treated as **token classification problem**
- Each word needs to have a certain label

Input: A textual snippet (typically a sentence)
Output: A label for each token in the sentence

1. **BIO** (**B**egin, **I**nside, **O**utside) (Sang ETK et al., 2000) ➔ CoNLL Shared Task (also found as IOB)

EGTA	inhibited	down	-	regulation	of	alpha	-	AR	mRNA	by	NE
B-Chemical	O	O	O	O	O	B-Protein	I-Protein	I-Protein	O	O	O

2. **IOE** (**I**nside, **O**utside, **E**nd)

EGTA	inhibited	down	-	regulation	of	alpha	-	AR	mRNA	by	NE
E-Chemical	O	O	O	O	O	I-Protein	I-Protein	E-Protein	O	O	O

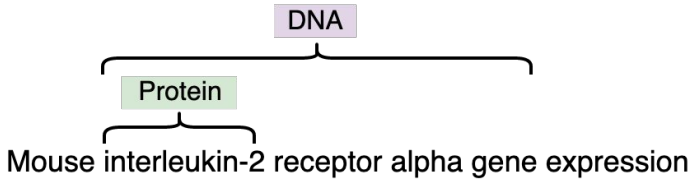
3. **IOBES** (**I**nside, **O**utside, **B**egin, **E**nd, **S**ingle) or **BILOU** (**B**egin, **I**nside, **L**ast, **O**utside, **U**nit)

EGTA	inhibited	down	-	regulation	of	alpha	-	AR	mRNA	by	NE
S-Chemical	O	O	O	O	O	B-Protein	I-Protein	E-Protein	O	O	O

NER: Usefulness & Challenges

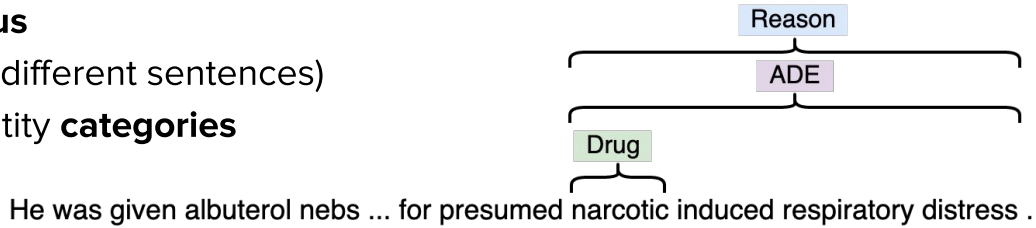
Usefulness:

- Initial component of many downstream tasks (Relation Extraction, QnA, etc)
- All existing applications use some kind of NER algorithm (recommendation systems, etc)
- Customer support ➜ find what the customer requests
- Classification ➜ based on contained entities



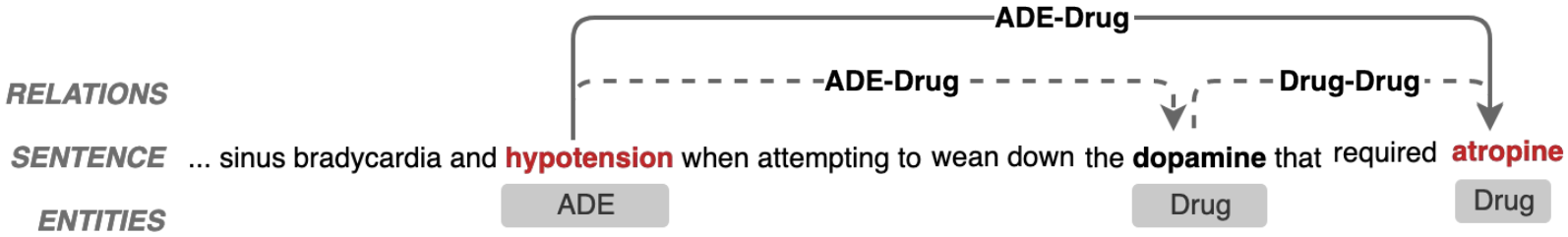
Challenges:

- Biomedical text contains a lot of **nested entities**
- Biomedical entities are very **ambiguous**
- Additional **context** required (e.g. from different sentences)
- Difficult to detect very **fine-grained** entity **categories**



(Textual) Relation Extraction

An **umbrella term** that describes the **identification of interactions between elements** in **text**

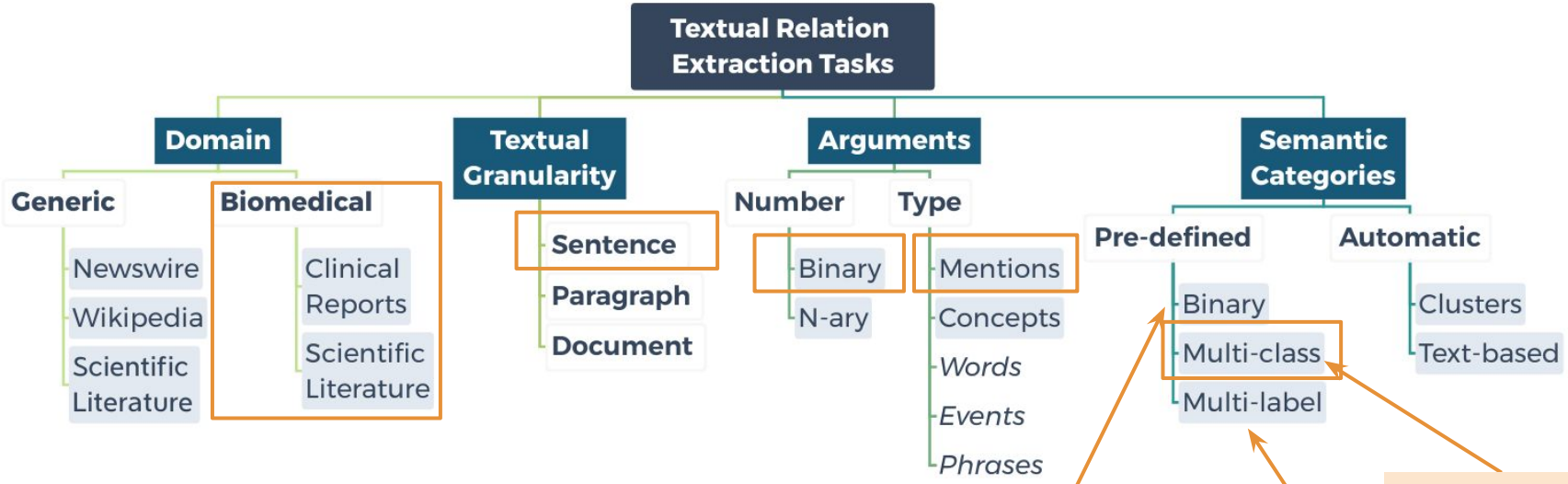


Input: A textual snippet (typically a sentence)
Output: A set of entity pairs with their relation label(s)

Definitions

- **Named Entity**: A word or a group of words that constitute a proper name (e.g. hypotension)
- **Entity Type**: Semantic category in which a named entity belongs (e.g. ADE, Drug)
- **Relation Type**: Semantic category of the relation (e.g. ADE-Drug)
- **Relation Argument**: Named entity that participates in a relation
- **Relation Instance**: A group of named entities participating in a relation along with their relation type

Relation Extraction: Taxonomy



- Multiple categorisations create multiple tasks
- Combinations of categories create other tasks
- **Most common**: Sentence-level Binary Relation Extraction

Relation Extraction: Usefulness

- Identification of patterns from raw text

Injury	disrupts	Physiological Function
Bodily Location	location-of	Biologic Function
Anatomical Structure	part-of	Organism
Pharmacologic Substance	causes	Pathological Function
Pharmacologic Substance	treats	Pathologic Function

- Creation of structured **Knowledge Bases**, useful for any application
- Automatic extraction of **complex biomedical associations**
- Important for **Question-Answering**
  - One can imagine that answering questions requires factual knowledge
  - Questions and answers can be formed as a relation extraction task (Lewis et al., 2019)

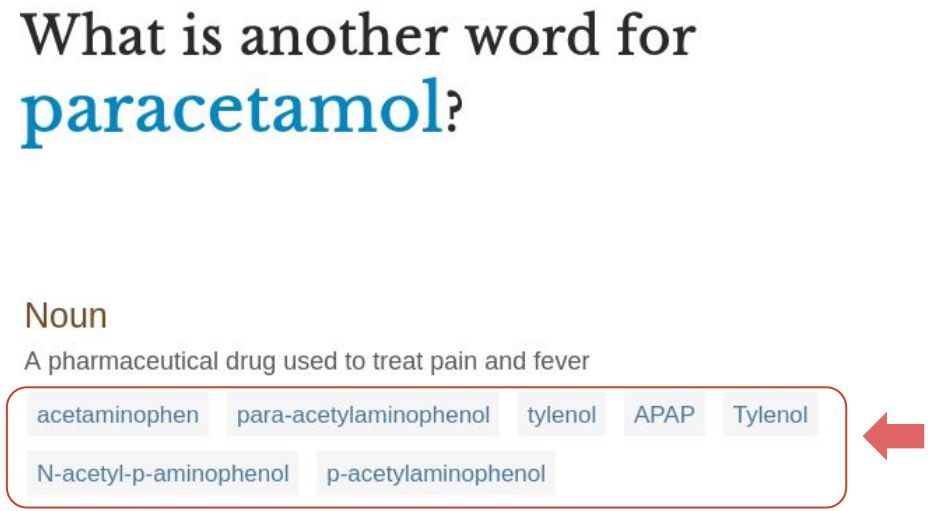
* Credits to: https://web.stanford.edu/class/cs124/lec/rel.pdf

Relation Extraction: Challenges

- Linguistic phenomena due to the diversity of language
- Grammatical phenomena (e.g. negation)
- Semantic challenges, e.g. implicit statements, common knowledge, etc

Relation Extraction: Challenges

- Named entities have many **aliases** - especially in the biomedical domain
  - Named Entity Linking / Normalisation is often used to resolve such cases



Relation Extraction: Challenges

- Named entities have many **aliases** - especially in the biomedical domain
  - Named Entity Linking / Normalisation is often used to resolve such cases
- Presence of hierarchies between entities: **Hyponymy** & **Hypernymy**
  - **Ontologies** can help identify such associations. E.g. WordNet

> **IS-A** (hypernym): Subsumption between classes
>
> Giraffe IS-A ruminant IS-A ungulate IS-A mammal IS-A animal ...

Relation Extraction: Challenges

- Named entities have many **aliases** - especially in the biomedical domain
  - Named Entity Linking / Normalisation is often used to resolve such cases
- Presence of hierarchies between entities: **Hyponymy** & **Hypernymy**
  - **Ontologies** can help identify such associations
- **Negation** alternates meaning and is expressed in various forms

> Neither ryanodine nor EGTA inhibited down-regulation of alpha-AR mRNA by NE.
>
> Further, AICAR pretreatment blocked PAR-1-induced increase in permeability of mouse-lung microvessels.

Relation Extraction: Challenges

- Named entities have many **aliases** - especially in the biomedical domain
  - Named Entity Linking / Normalisation is often used to resolve such cases
- Presence of hierarchies between entities: **Hyponymy** & **Hypernymy**
  - **Ontologies** can help identify such associations
- **Negation** alternates meaning and is expressed in various forms
- Detecting **distant arguments** is difficult
  - E.g. intermediate parenthetical phrases

The case of a 40-year-old patient with **bilateral optic neuropathy**, who underwent unsuccessful cadaver kidney transplantation surgery, was treated with ethambutol and **isoniazid**.

Relation Extraction: Challenges

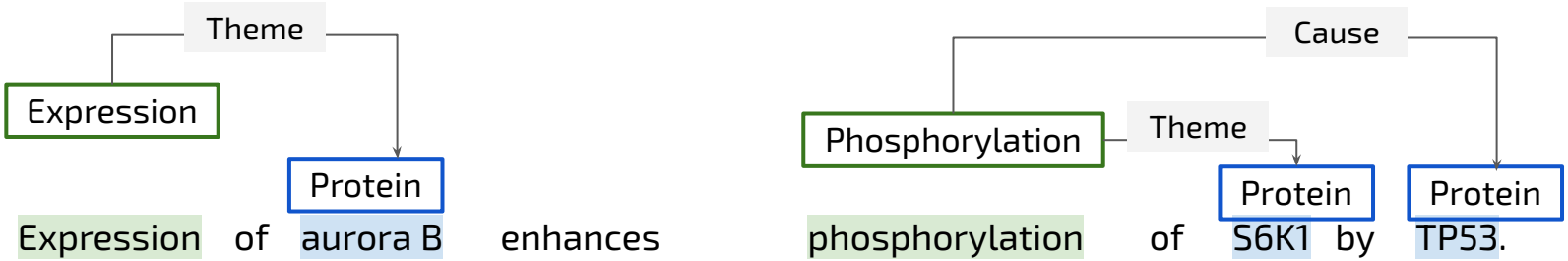
- Named entities have many **aliases** - especially in the biomedical domain
  - Named Entity Linking / Normalisation is often used to resolve such cases
- Presence of hierarchies between entities: **Hyponymy** & **Hypernymy**
  - **Ontologies** can help identify such associations
- **Negation** alternates meaning and is expressed in various forms
- Detecting **distant arguments** is difficult
  - E.g. intermediate parenthetical phrases
- Discourse contains a lot of **implicit relations, temporal relations**

Allergies: Bactrim (rash)

The patient had two transfusion reactions to platelets when first admitted . She was premedicated with antihistamines .

Event Extraction

n-ary relations combined



Event E1

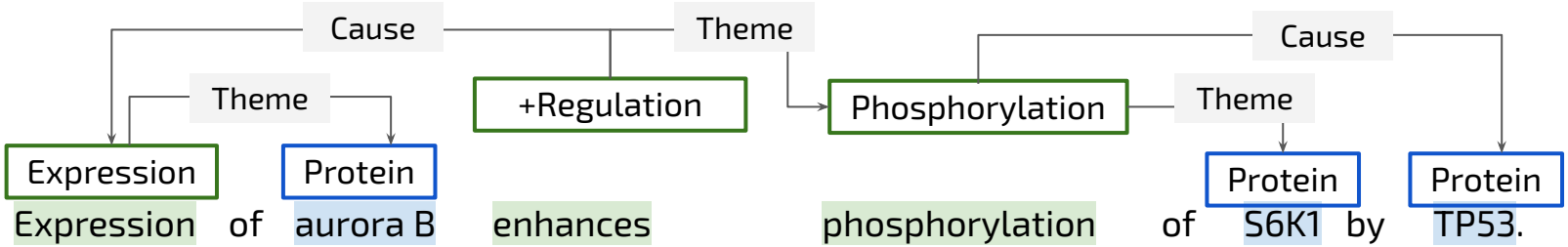
Type: Gene_expression
Trigger: *Expression*
Theme: *aurora B*

Event E2

Type: Phosphorylation
Trigger: *phosphorylation*
Theme: *S6K1*
Cause: *TP53*

Event Extraction

n-ary relations combined and **nested**



Event E1

Type: Gene_expression
Trigger: *Expression*
Theme: *aurora B*

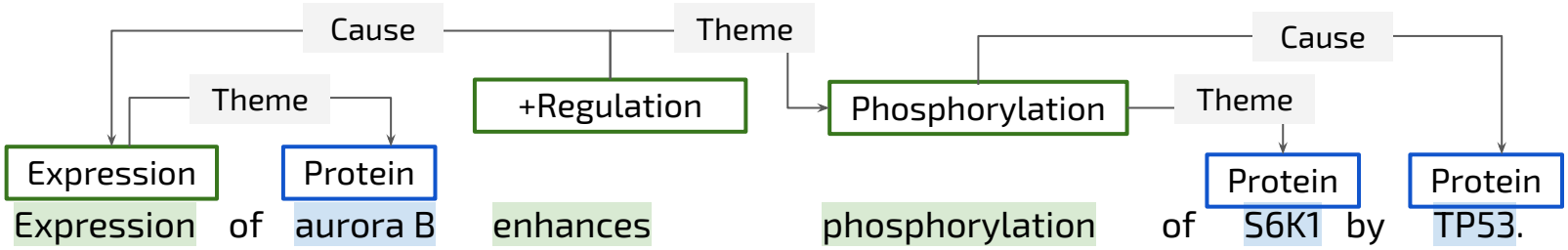
Event E2

Type: Phosphorylation
Trigger: *phosphorylation*
Theme: *S6K1*
Cause: *TP53*

Event E3

Type: Positive_regulation
Trigger: *enhances*
Cause: **E1**
Theme: **E2**

Event Anatomy

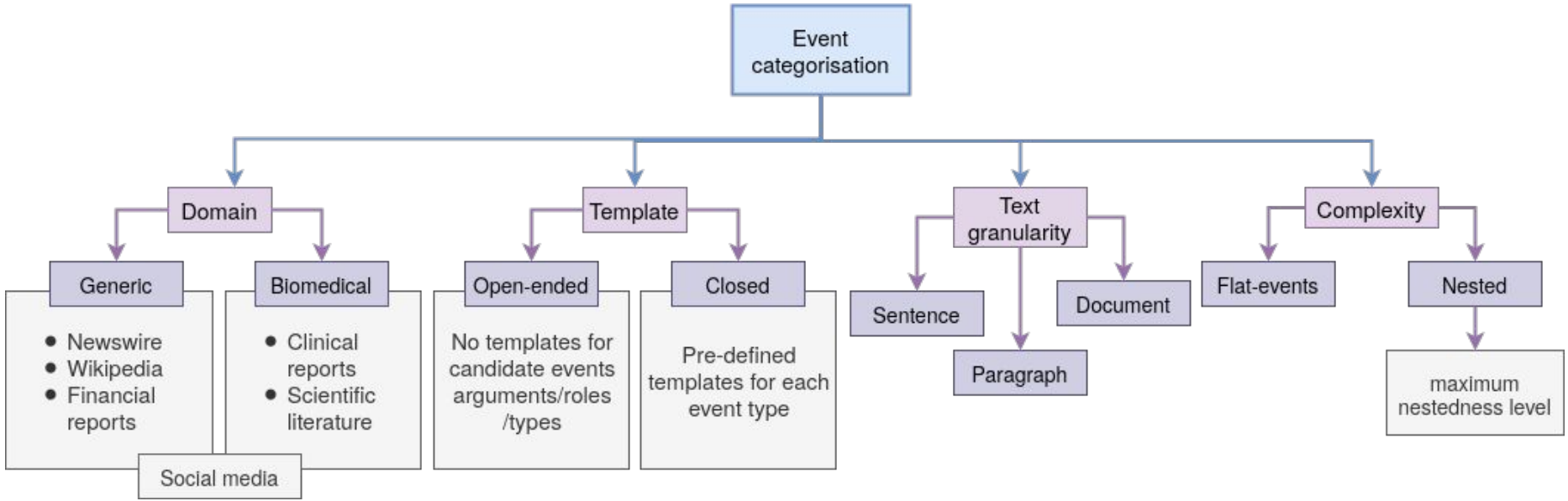


Event mention: The words comprising the event in a document
Event type: The class label of an event
Event trigger: The word(s) that expresses the event mention and indicates its **type**
Event argument(s): Entity mention or other event that serves as a **participant** or attribute
Argument role: The **relationship** between an argument and the event

Sample Event Template

Type: Phosphorylation
Trigger type: <*Phosphorylation*>
Argument 1 Type: <Theme>
Argument 1 NEs: <Protein | Expression>
Argument 2 Type: <Cause>
Argument 2 NEs: <Protein | Drug>

Event Extraction categories



Event Extraction: Usefulness

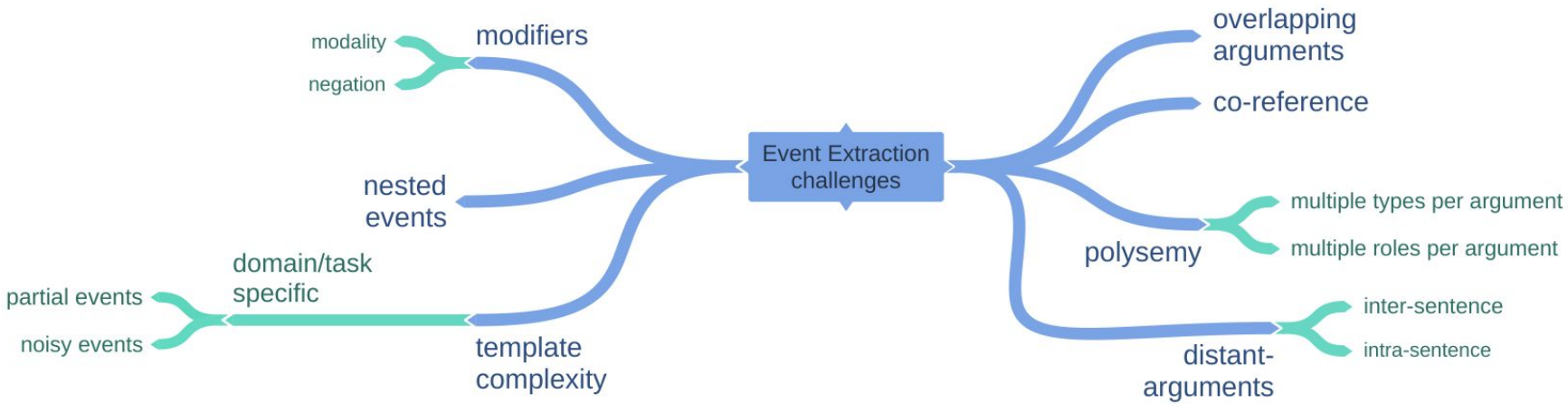
Why do we need them:

- Events are structured information units:
  - Structured Information Extraction
  - Faceted search
  - Question answering
- Event structure can be mapped to biomolecular interaction structure
  - Constructing (extending) biomolecular networks
  - Semi-automated curation of pathways
- Linking of event mentions across texts can help inferring new associations
  - Knowledge discovery
  - Hypothesis formulation

Impacted areas:

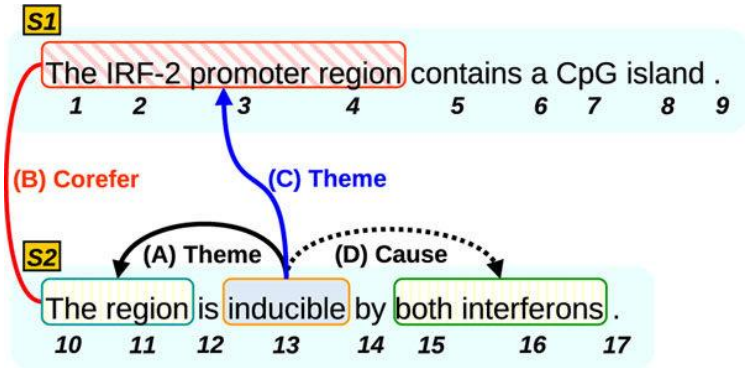
- Pharmacogenetics
  - New drug-protein interactions
- Translational genomics
  - Genome-based tests for standard clinical care
- Cancer research
  - Pathway construction and simulations
- Systems biology
- Functional genomics

Event Extraction: Challenges



Event Extraction: Challenges

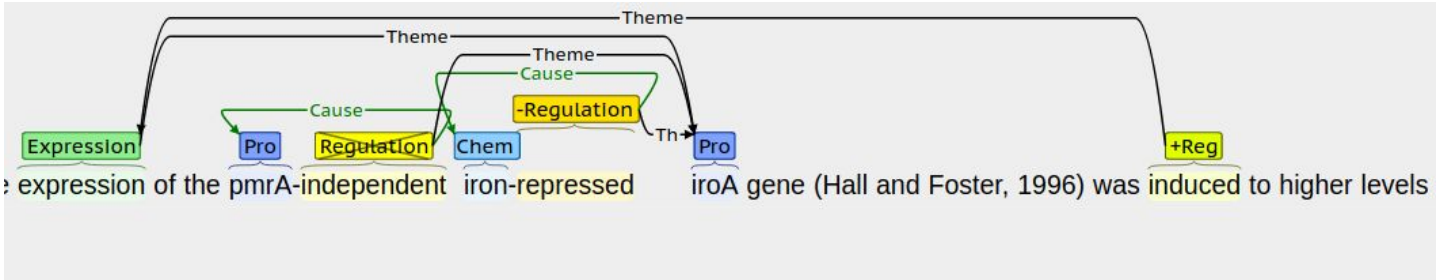
- **Coreference**: The entity is represented by a **pronoun** or **abstract mention**.
  The actual entity/trigger representation is mentioned in another sub-phrase, sentence



* **Credits to:** Katsumasa Yoshikawa et al., 2011
https://jbiomedsem-biomedcentral-com.manchester.idm.oclc.org/articles/10.1186/2041-1480-2-S5-S6

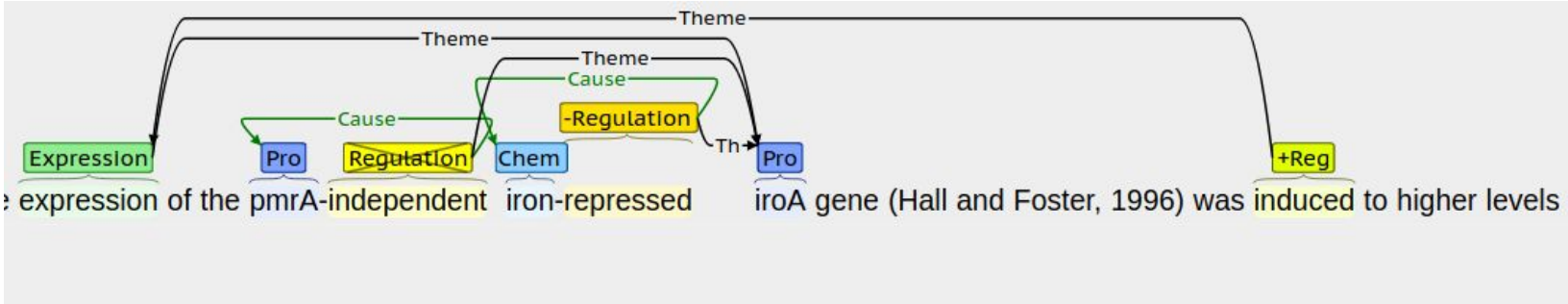
Event Extraction: Challenges

- **Coreference**: The entity is represented by a **pronoun** or **abstract mention**.
  The actual entity/trigger representation is mentioned in another sub-phrase, sentence.
- An event trigger might **overlap fully or partially** with another event trigger, or event an entity. **Tokenization** might affect the event processing



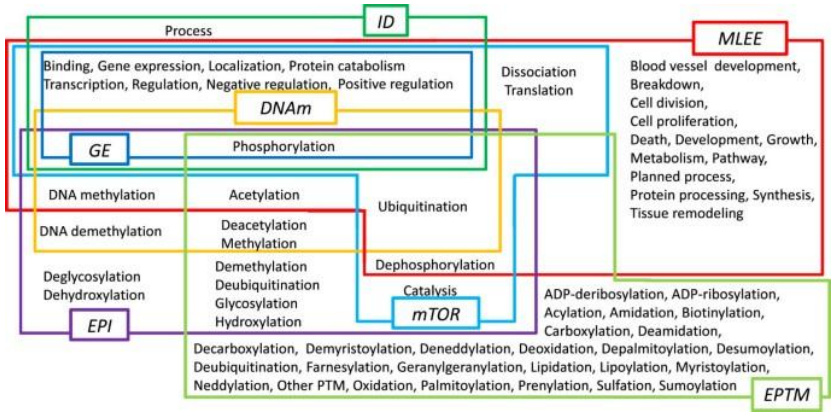
Event Extraction: Challenges

- **Coreference**: The entity is represented by a **pronoun** or **abstract mention**.
  The actual entity/trigger representation is mentioned in another sub-phrase, sentence.
- An event trigger might **overlap fully or partially** with another event trigger, or event an entity. **Tokenization** might affect the event processing
- An event can (a) be an argument to other events (b) also have other events as arguments. The **depth of the underlying event graph** can expand rapidly!



Event Extraction: Challenges

- **Coreference**: The entity is represented by a **pronoun** or **abstract mention**.
  The actual entity/trigger representation is mentioned in another sub-phrase, sentence.
- An event trigger might **overlap fully or partially** with another event trigger, or event an entity. **Tokenization** might affect the event processing
- An event can (a) be an argument to other events (b) also have other events as arguments. The **depth of the underlying event graph** can expand rapidly!
- **Generalisation** to other datasets within the domain can be challenging, due to **clashing event structures.**



Event Extraction: Challenges

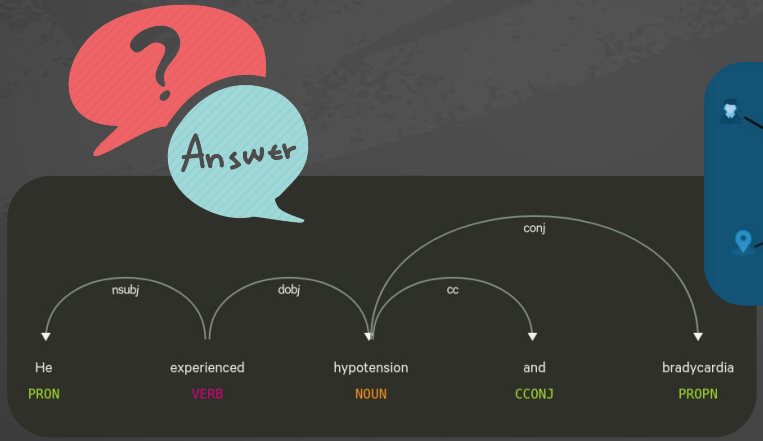
- **Coreference**: The entity is represented by a **pronoun** or **abstract mention**.
  The actual entity/trigger representation is mentioned in another sub-phrase, sentence.
- An event trigger might **overlap fully or partially** with another event trigger, or event an entity. **Tokenization** might affect the event processing
- An event can (a) be an argument to other events (b) also have other events as arguments. The **depth of the underlying event graph** can expand rapidly!
- **Generalisation** to other datasets within the domain can be challenging, due to **clashing event structures.**
- Extracting the event mention might be insufficient: context (meta-knowledge) matters!

BRAF affects the binding of MUC1 to PKM2.
BRAF **does not** affect the binding of MUC1 to PKM2.

Event Extraction: Challenges

- **Coreference**: The entity is represented by a **pronoun** or **abstract mention**.
  The actual entity/trigger representation is mentioned in another sub-phrase, sentence.
- An event trigger might **overlap fully or partially** with another event trigger, or event an entity. **Tokenization** might affect the event processing
- An event can (a) be an argument to other events (b) also have other events as arguments. The **depth of the underlying event graph** can expand rapidly!
- **Generalisation** to other datasets within the domain can be challenging, due to **clashing event structures.**
- Extracting the event mention might be insufficient: context (meta-knowledge) matters!
- The same event type might have **different argument roles** and/or **entity types** in each sentence

Additional BioNLP Tasks



Metaknowledge Identification

Are all event mentions the same?

BRAF affects the binding of MUC1 to PKM2.

BRAF **does not** affect the binding of MUC1 to PKM2.

We **examined how** BRAF affects the binding of MUC1 to PKM2.

We **prove that** BRAF affects the binding of MUC1 to PKM2.

We **hypothesize that** BRAF affects the binding of MUC1 to PKM2.

BRAF affects the binding of MUC1 to PKM2 **[4]**.

Metaknowledge Identification

Are all event mentions the same?

BRAF affects the binding of MUC1 to PKM2.

- Contextual information within the same sentence
- Modifies the event
- Additional event classification layers

Based on **our experiments** MUC1 binds **weakly** to PKM2

Knowledge type: Observation — Manner: Low

We wanted to **investigate whether** MUC1 binds to PKM2

Knowledge type: Investigation → Certainty: Low

According to Wong et al, (2014) MUC1 **cannot** bind to PKM2

Source: Other — Polarity: Negative

Entity Normalisation

Is TP53 same as p53? What does this mean?

Also known as:

- Named Entity Linking
- Named Entity Disambiguation

Task: assign **unique entity identifiers** to each NE, based on one or more **reference** resources

Reference resources:

- Dictionaries, Thesauri
  - **MeSH terms:** Controlled and hierarchically-organized vocabulary
- Ontologies
  - **ChEBI:** Ontology of molecular entities focused on 'small' chemical compounds.
  - **BioPortal:** Combination of >300 ontologies
- Knowledge bases - Databases
  - **UniProt:** A database of protein sequence and functional information
  - **UMLS:** A structured combination of >1M biomedical concepts from over 100 source vocabularies
    - Includes the Semantic Network of semantically organised concepts
  - **GENBank | DRUGBank**

Entity Normalisation

Process:

NER

Candidate generation

Disambiguation

text-based | graph-based

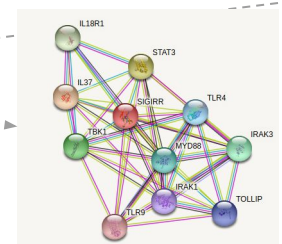
Link with additional context

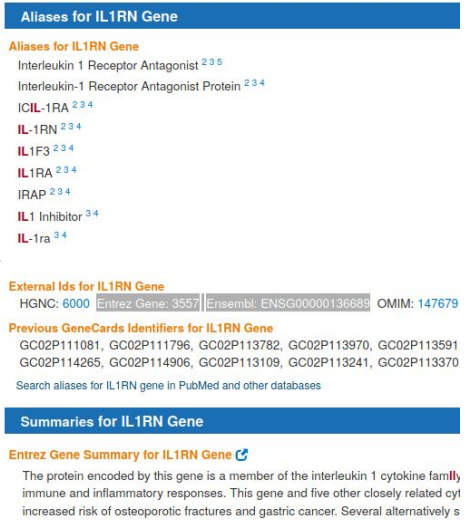
SARS-CoV was able to induce greater IL-1R when compared to Influenza-A.

UniProtKB: P14778	Interleukin 1 Alpha
UniProtKB: P05231	Interleukin 1 Receptor Antagonist
UniProtKB: P18510	Interleukin 1 Receptor Type 1
...	...

Interleukin 1 Receptor Antagonist

- UniProtKB: P14778
- Entrez Gene: 3557
- Ensembl: ENSG00000136689





Question Answering

Given a query retrieve relevant information

Also known as:

- Information retrieval
- Document retrieval

Shared Tasks

TREC

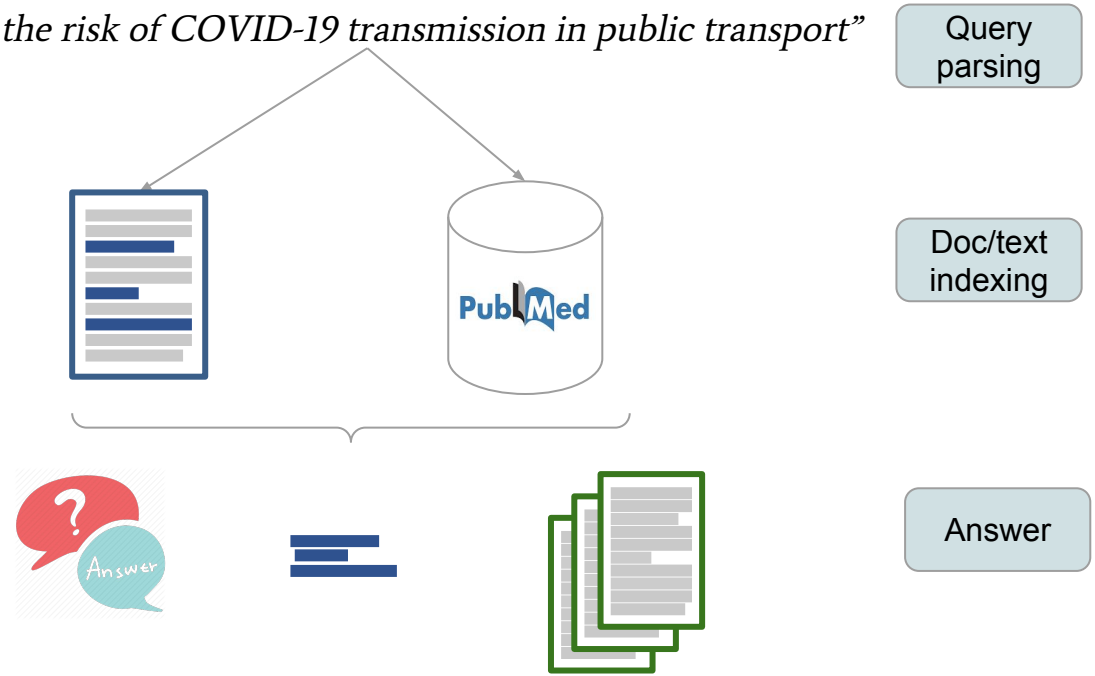
- Genomics
- LiveQA

CLEF

- BioASQ

Kaggle

- CORD-19 challenge



Text Summarisation

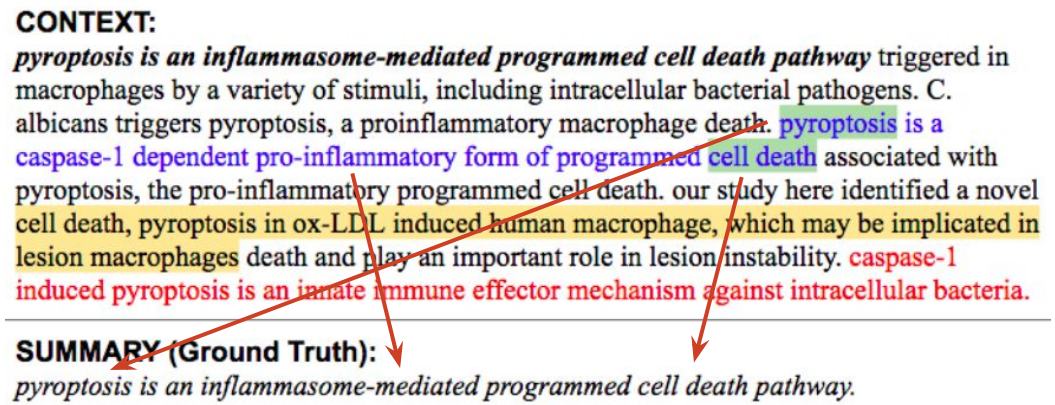
➜ The task that aims to produce a reduced version of a given snippet

Input: A textual snippet
Output: A summary of it

Methods:

- **Extractive**:
  Extract pieces of text and glue them together
  - The most common and easiest task

- **Abstractive**:
  Generate a new summary
  - Sequence-to-sequence models are incorporated
  - Grammaticality is important



Applications

Applications

NaCTeM Tools
A large list can be found here: http://nactem.ac.uk/software.php & http://www.nactem.ac.uk/services.php

- **Thalia** (http://www.nactem.ac.uk/Thalia/)
  - A semantic search engine for Pubmed abstracts
- **LitPathExplorer** (http://nactem.ac.uk/LitPathExplorer_BI/)
  - A confidence-based visual text analytics tool for exploring literature-enriched pathway models
- **History of Medicine** (http://www.nactem.ac.uk/MHM/)
  - A semantic search system over historical medical archives
- **RobotAnalyst** (http://www.nactem.ac.uk/robotanalyst/)
  - A tool to minimise the human workload involved in the study identification phase of systematic reviews

Thalia

Text mining for **H**ighlighting, **A**ggregating and **L**inking **I**nformation in **A**rticles

http://nactem.ac.uk/Thalia_BI/

Semantic Search Engine: Recognition of named entities appearing in biomedical abstracts (PubMed)

- 8 types of concepts:
  - Chemicals
  - Diseases
  - Drugs
  - Genes
  - Metabolites
  - Proteins
  - Species
  - Anatomical entities

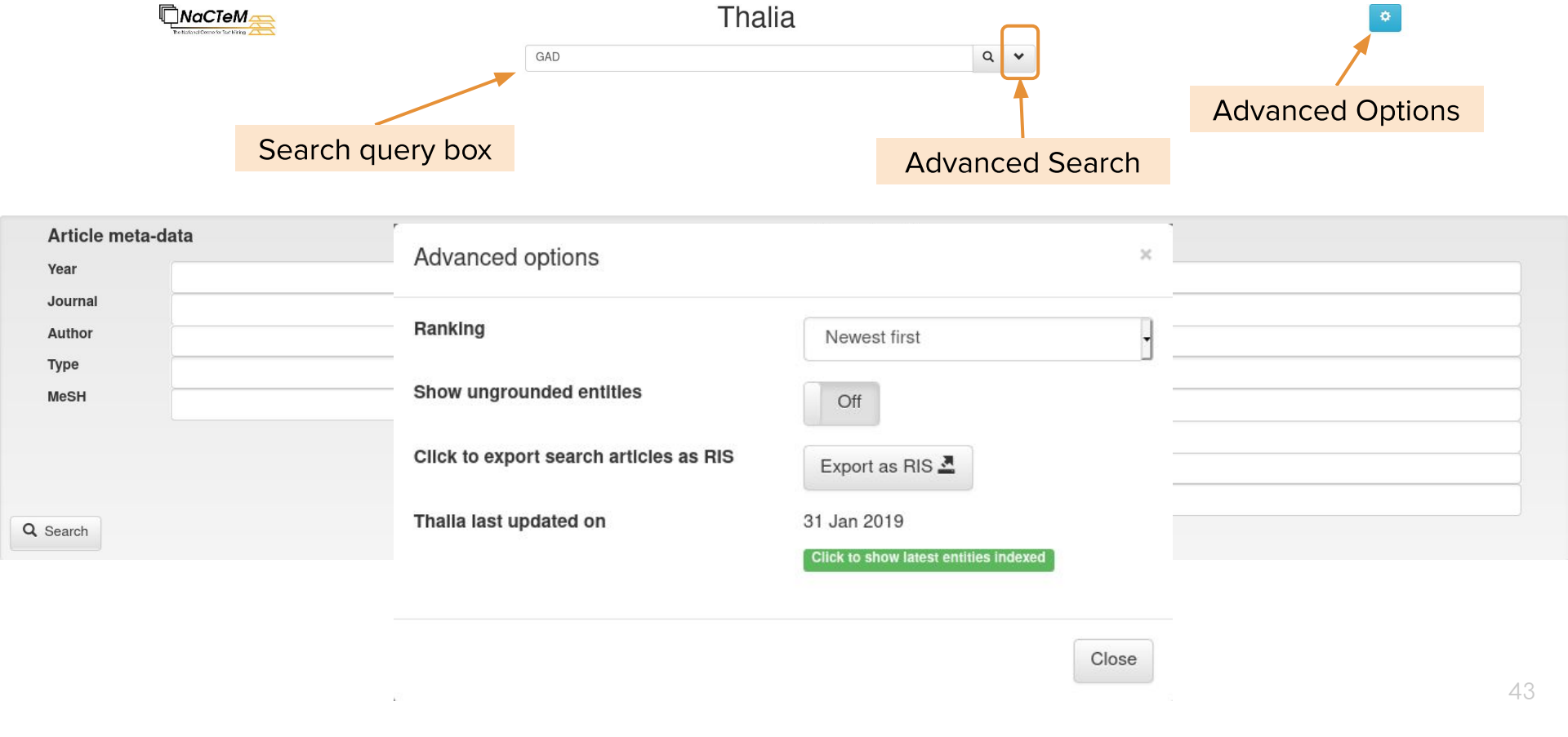
http://nactem.ac.uk/Thalia/

Publications

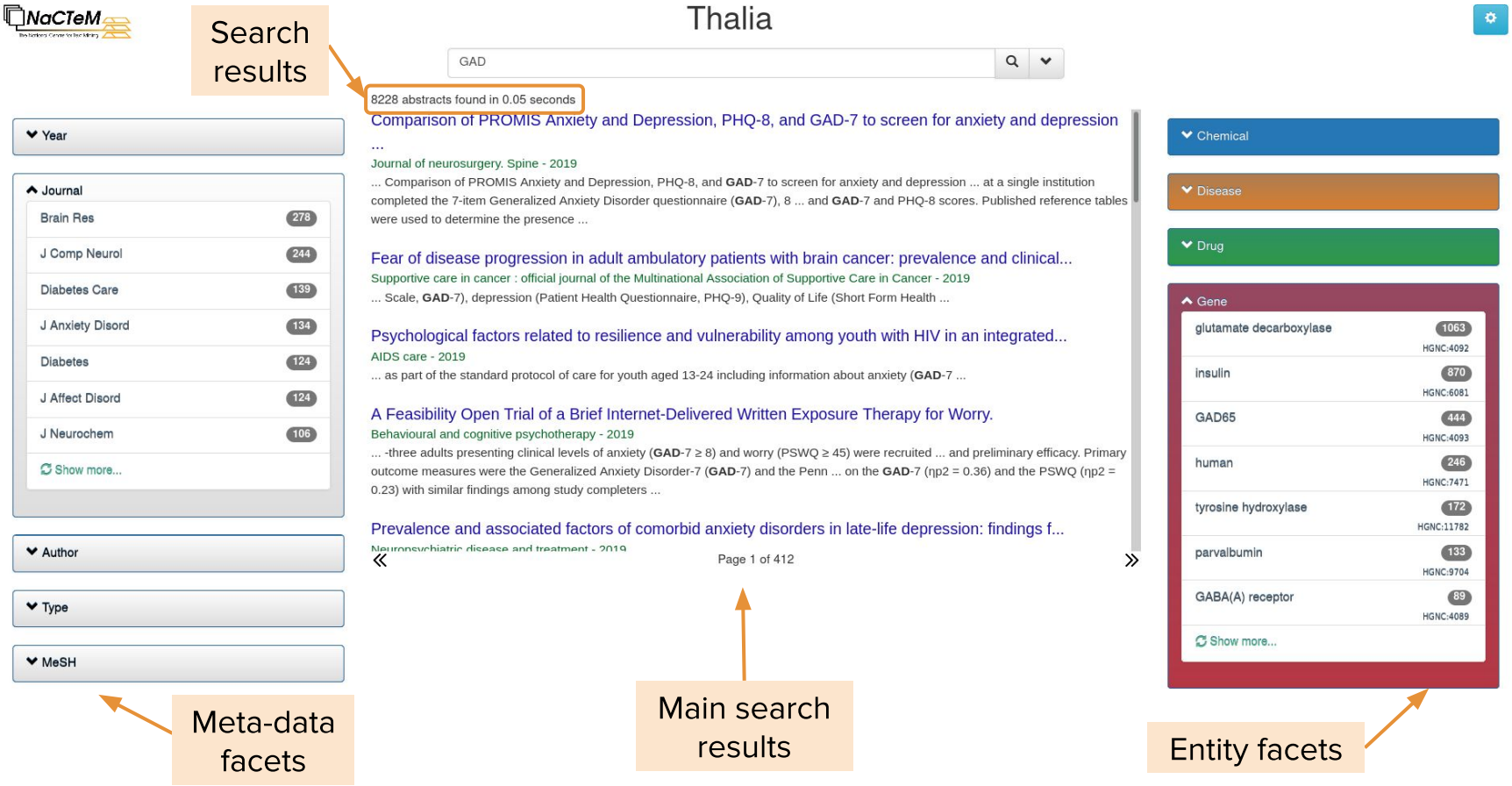
Axel J Soto, Piotr Przybyła and Sophia Ananiadou (2018). Thalia: Semantic search engine for biomedical abstracts. *Bioinformatics*.

Piotr Przybyła, Axel J. Soto, and Sophia Ananiadou (2017). Identifying Personalised Treatments and Clinical Trials for Precision Medicine using Semantic Search with Thalia, In *Proceedings of the Twenty-Fifth Text REtrieval Conference (TREC)*.

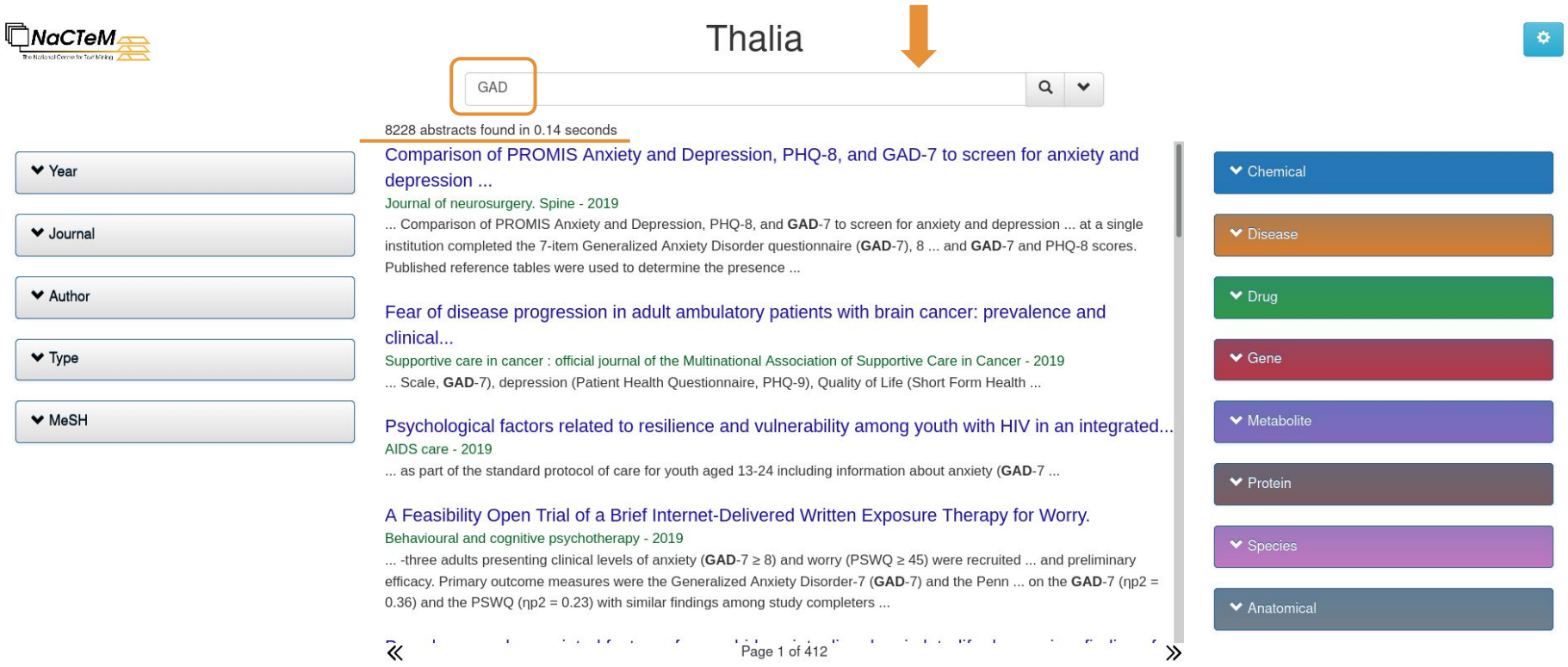
Thalia: General Infrastructure



Thalia: General Infrastructure

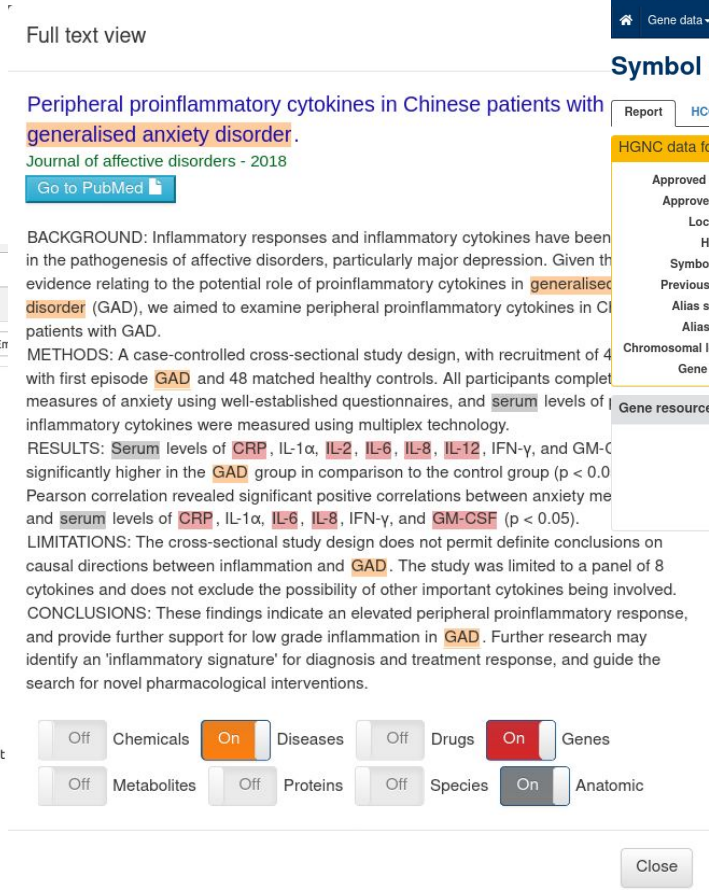


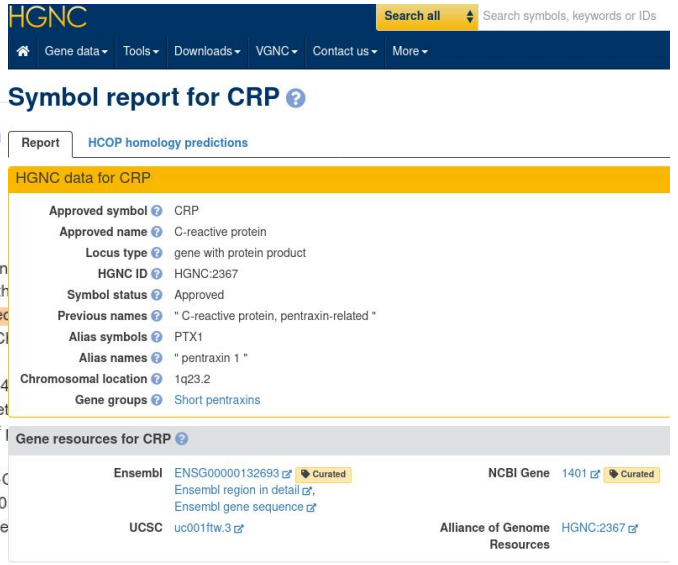
Thalia: Querying



Thalia: Querying

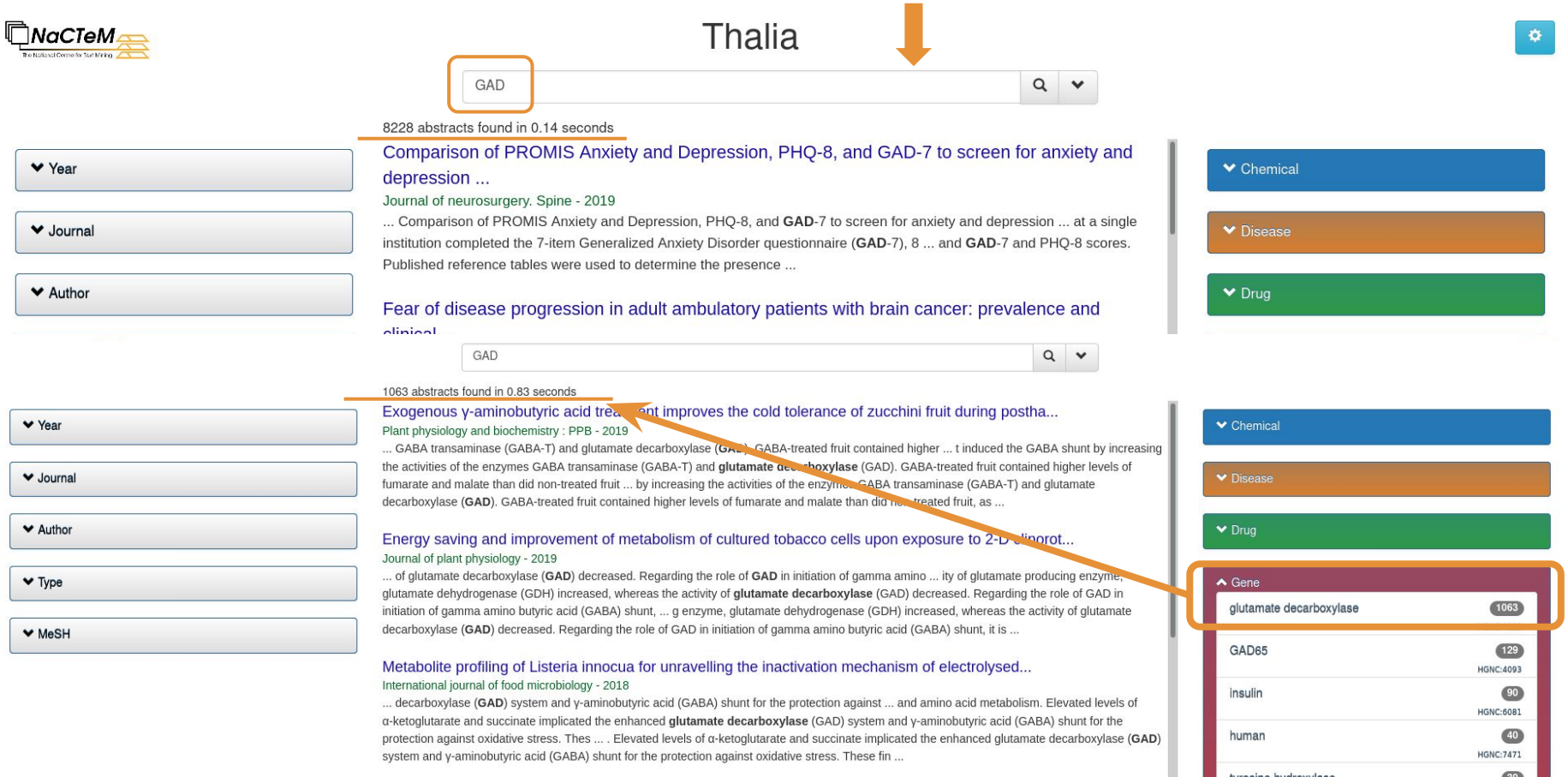
We can access the PubMed article



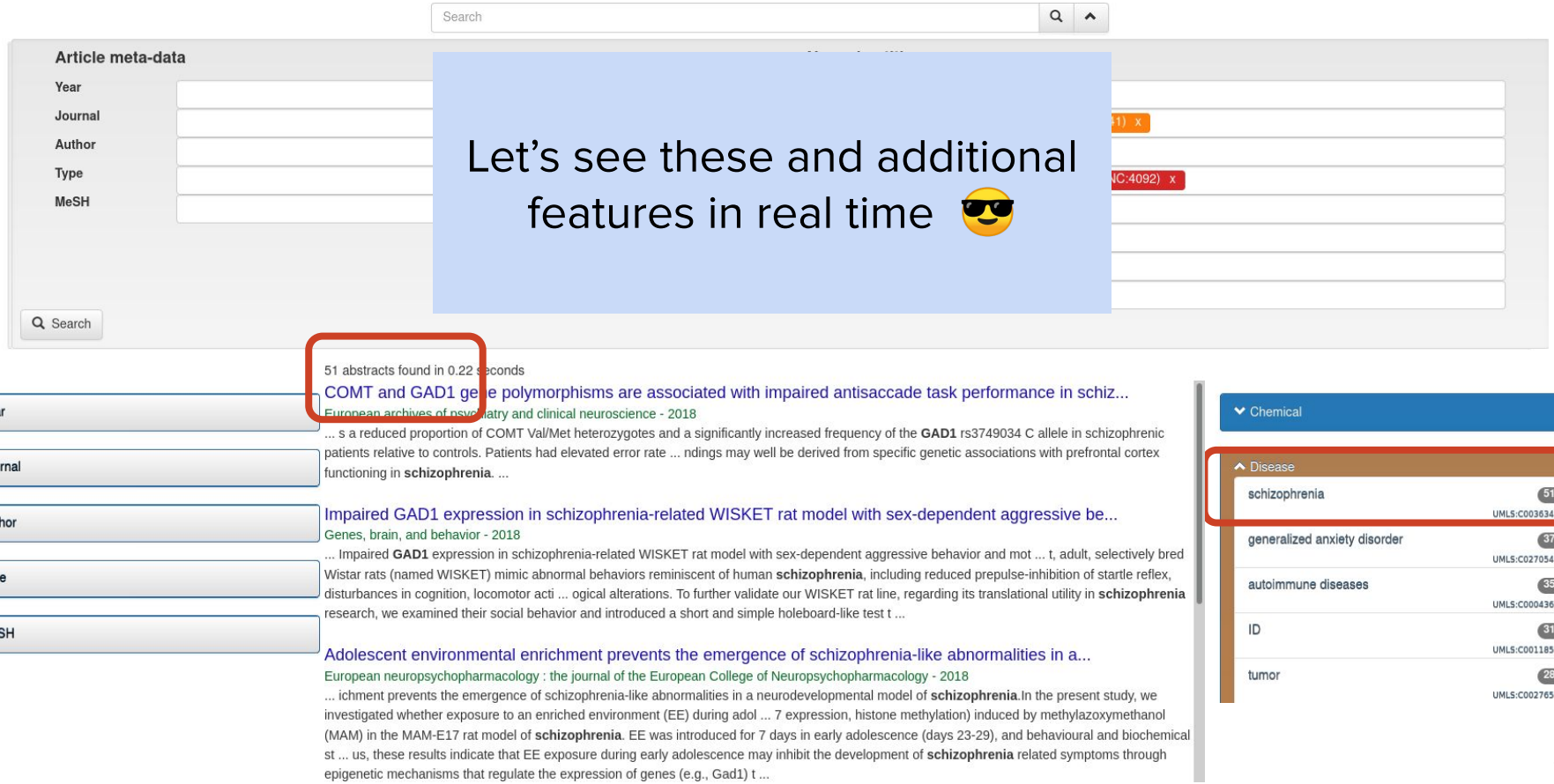


We can get information about an entity

Thalia: Filtering based on entity types



Thalia: Finding co-occurring entities

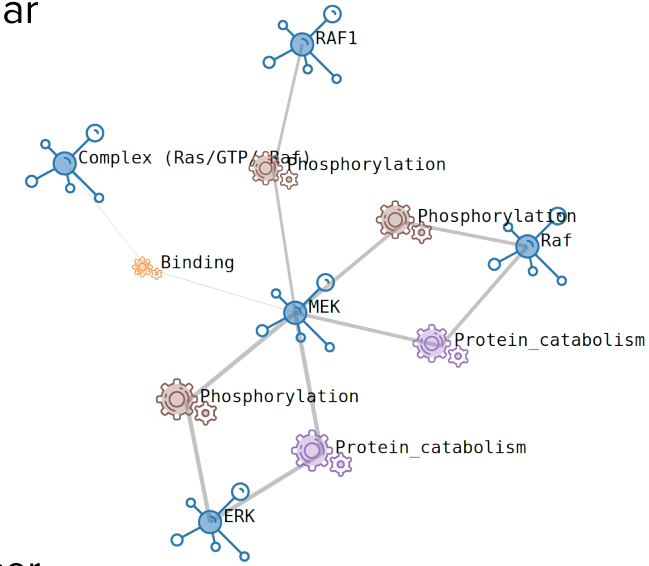


LitPathExplorer

http://nactem.ac.uk/LitPathExplorer_BI/

Bio-events in use: Enhancing interactions in biomolecular pathway networks

1. **Flexible** search and exploration
   - multi-view
   - **interactive** functionalities
2. Identification and visualisation of corroborating/contradicting **textual evidence**
3. **Discovery** and integration of new interactions
4. **Human-in-the-loop** analytical process - adapting to user needs



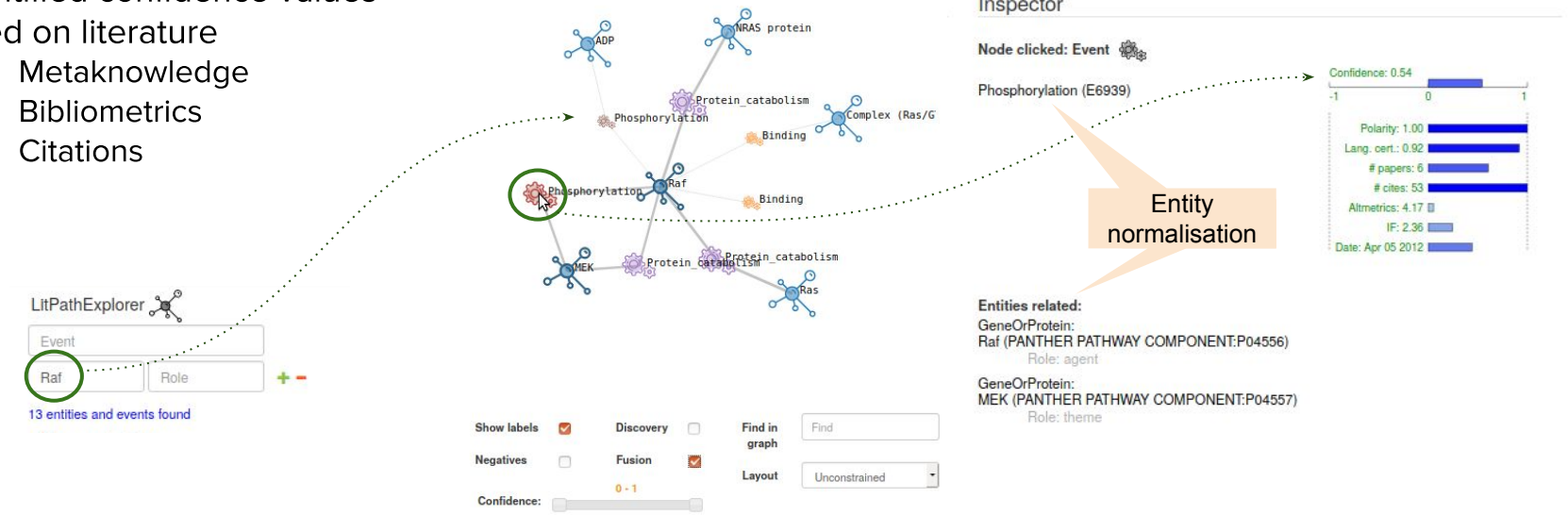
LitPathExplorer: Contextualisation

Mapping pathway interactions to textual evidence requires a TM pipeline:

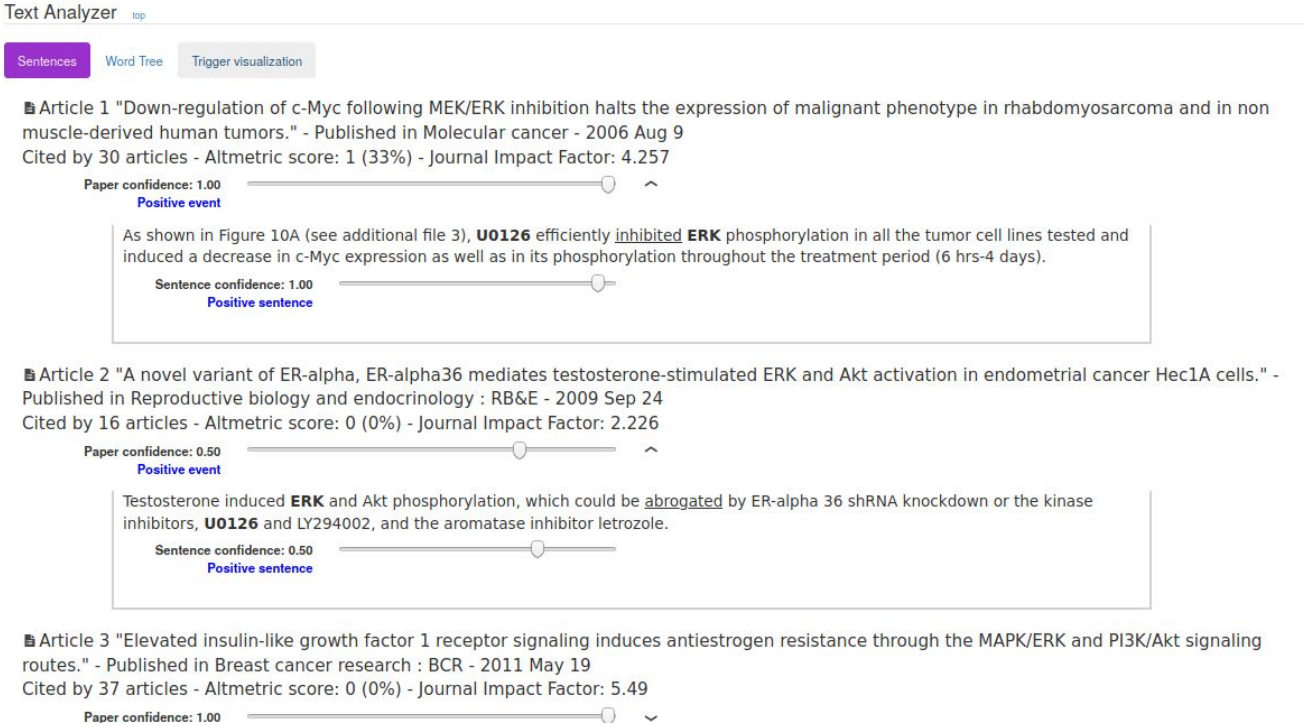
➔ NER ➔ EE ➔ NE Normalisation ➔ Metaknowledge ➔ Bibliometrics➔ Event Mapping

Quantified confidence values based on literature

- Metaknowledge
- Bibliometrics
- Citations



LitPathExplorer: Textual evidence



Textual confidence:

- Polarity + uncertainty
- Multi-level
  - Sentence
  - Paper

Interactive-adaptive:

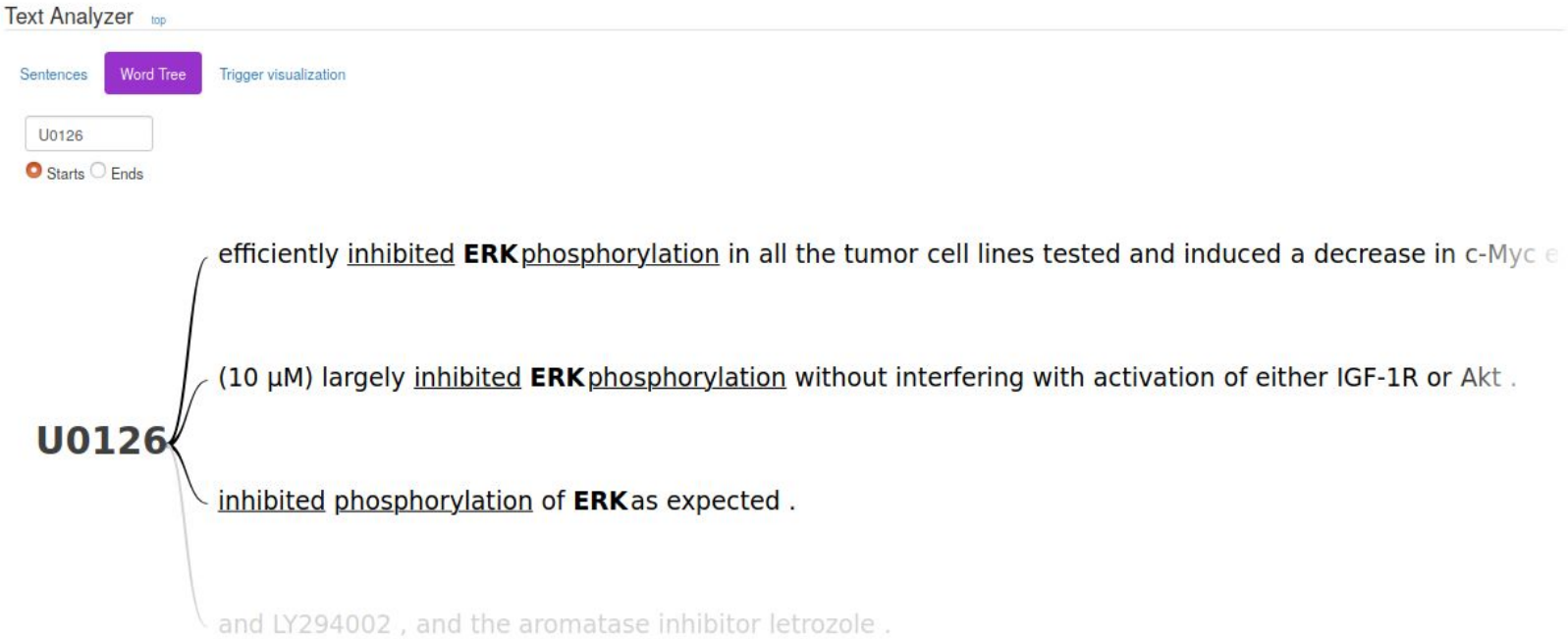
- User feedback **propagated** (upwards aggregation)
- Used as data for **online learning** (2 layer NN)

LitPathExplorer: Textual evidence

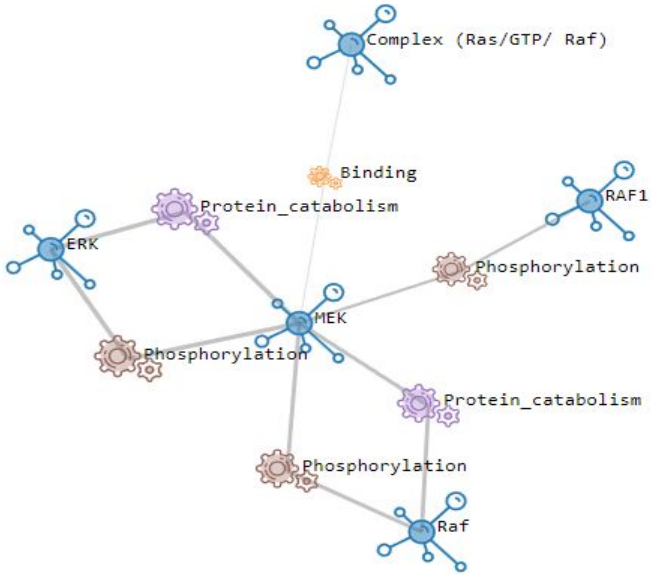
Alternative views:

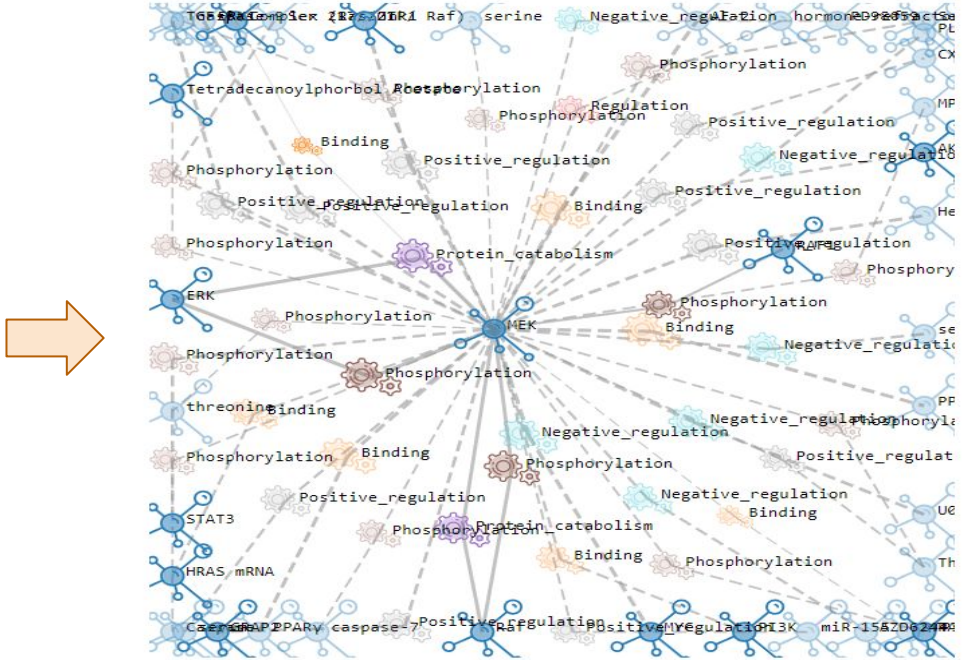
LitPathExplorer: Textual evidence

Alternative views:



LitPathExplorer: Discovery





Show labels ☑ Discovery ☐ Find in graph Find

Negatives ☐ Fusion ☑ Layout Unconstrained

Confidence: 0 - 1

filter

Show labels ☑ Discovery ☑ Find in graph Find

Negatives ☐ Fusion ☑ Layout Unconstrained

Confidence: 0 - 1

History of Medicine

Discover and extract information automatically from medical historical archives

- Archives:
  - The British Medical Journal (BMJ) (1840-2013)
  - The London Medical Officer of Health reports (MOH) (1848-1972)
- **Bibliographic information**
  - Author name, publication date, index (BMJ or MOH), journal
- **Semantic Information**
  - **Entities**: medical conditions, signs or symptoms, environmental factors
  - **Events**: Relationships between entities
- **Suggestions of related query terms**
- **Historical tracking of query terms**

http://nactem.ac.uk/MHM/

http://www.nactem.ac.uk/hom/

Publications

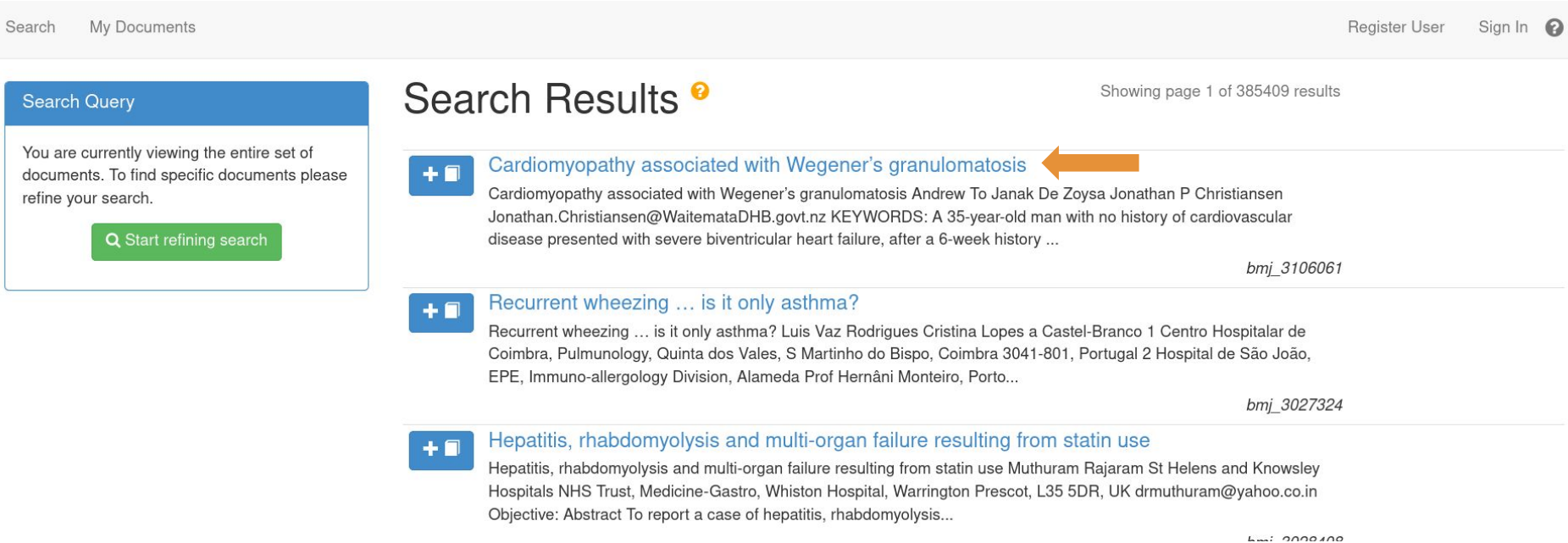
Thompson, P., Batista-Navarro, R. T. B., Kontonatsios, G., Carter, J., Toon, E., McNaught, J., Timmermann, C., Worboys, M. and Ananiadou, S. (2016). Text Mining the History of Medicine. *PLOS One*.

Thompson, P., Carter, J., McNaught, J. and Ananiadou, S. (2015). Semantically Enhanced Search System for Historical Medical Archives. In *Proceedings of Digital Heritage*.

Thompson, P., McNaught, J. and Ananiadou, S. (2015). Customised OCR Correction for Historical Medical Text. In *Proceedings of Digital Heritage*.

Miwa, M., Thompson, P., Korkontzelos, I. and Ananiadou, S. (2014). Comparable Study of Event Extraction in Newswire and Biomedical Domains. In *Proceedings of Coling*.

History of Medicine: Documents



History of Medicine: Documents

Search My Documents Register User Sign In

Cardiomyopathy associated with Wegener's granulomatosis

Showing page 1 of 385409 results

Record

Reference
3106061

Title
Cardiomyopathy associated with Wegener's granulomatosis

Year
2009

Page Info
bcr2006098038

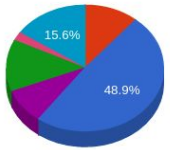
Index
bmj

Content

Due to licensing conditions we are unable to provide the full text of this document, however the article is available externally in the following repositories.

- Europe PubMed Central
- PubMed Central

Entity Types



Entities

- ▶ Anatomical (5)
- ▶ Condition (22)
- ▶ Environmental (4)
- ▶ Sign or Symptom (6)
- ▶ Subject (1)
- ▶ Therapeutic or Investigational (7)

Events

- ▶ Affect (2)

atosis

To Janak De Zoysa Jonathan P Christiansen
year-old man with no history of cardiovascular
ek history ...

bmj_3106061

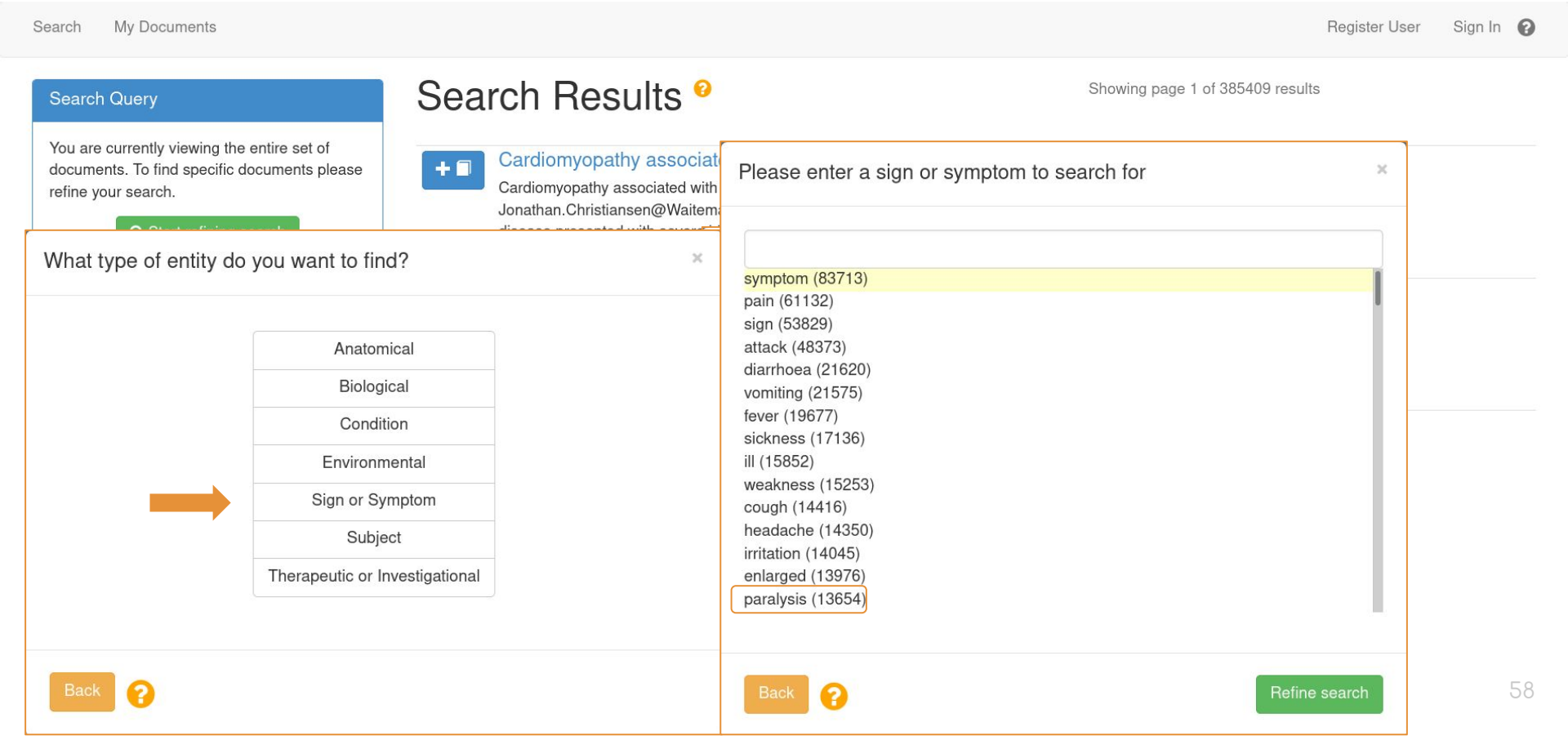
a Lopes a Castel-Branco 1 Centro Hospitalar de
nbra 3041-801, Portugal 2 Hospital de São João,
Porto...

bmj_3027324

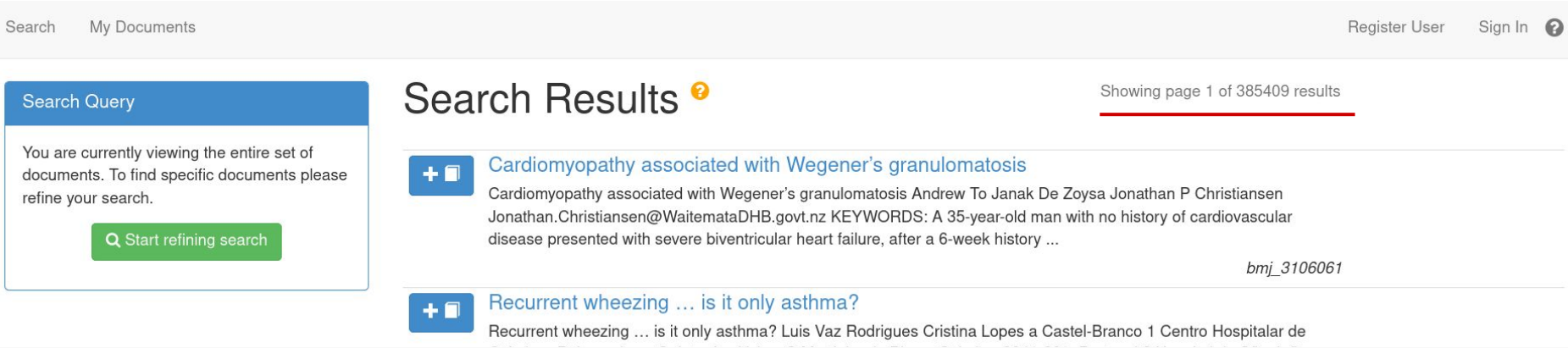
lting from statin use

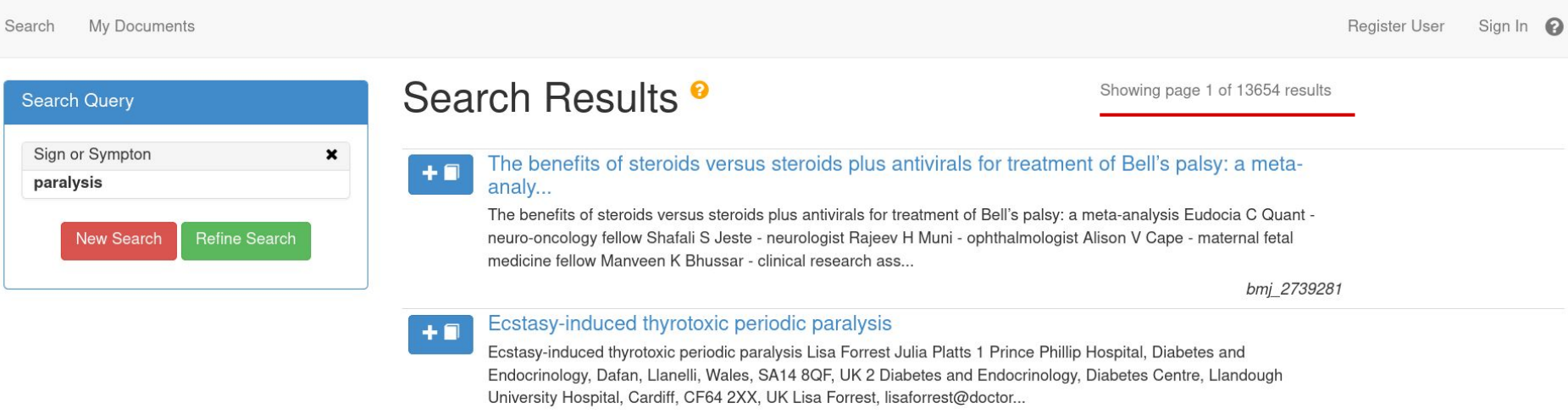
use Muthuram Rajaram St Helens and Knowsley
Prescot, L35 5DR, UK drmuthuram@yahoo.co.in

History of Medicine: Search Entities

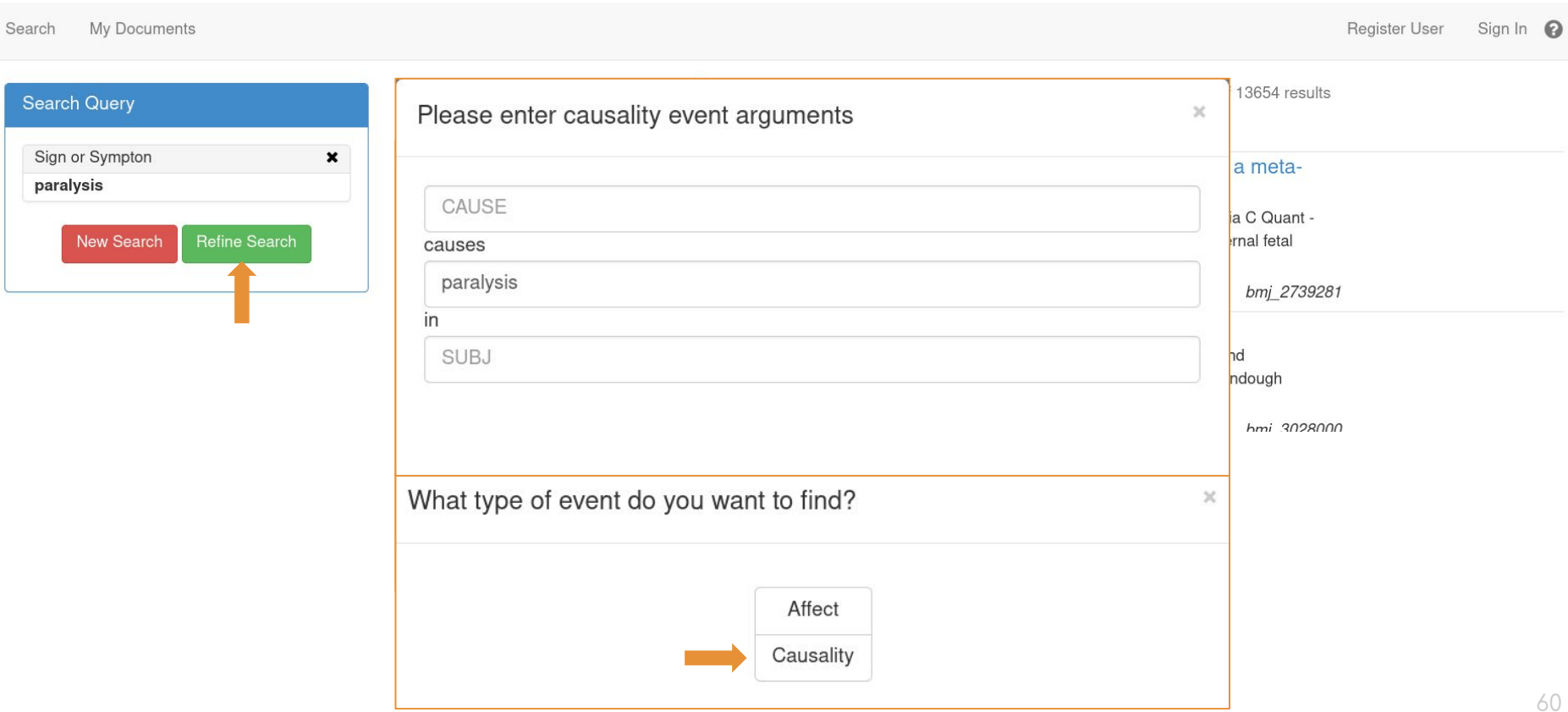


History of Medicine: Search Entities

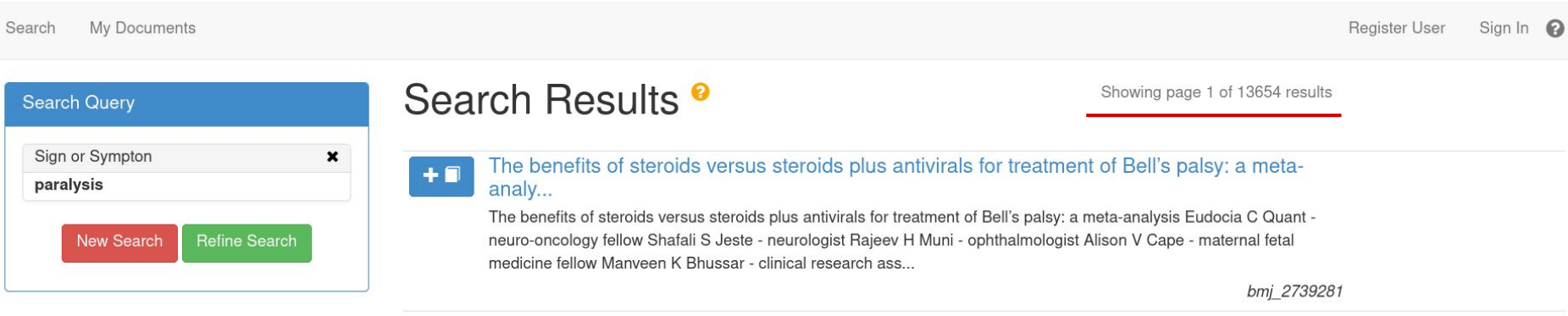


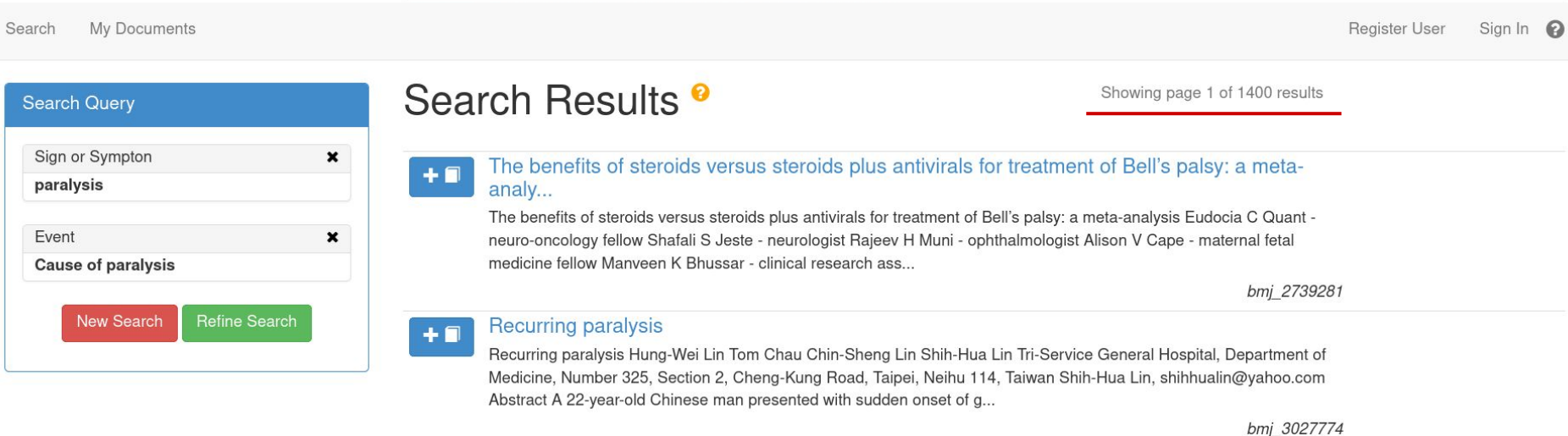


History of Medicine: Search Events

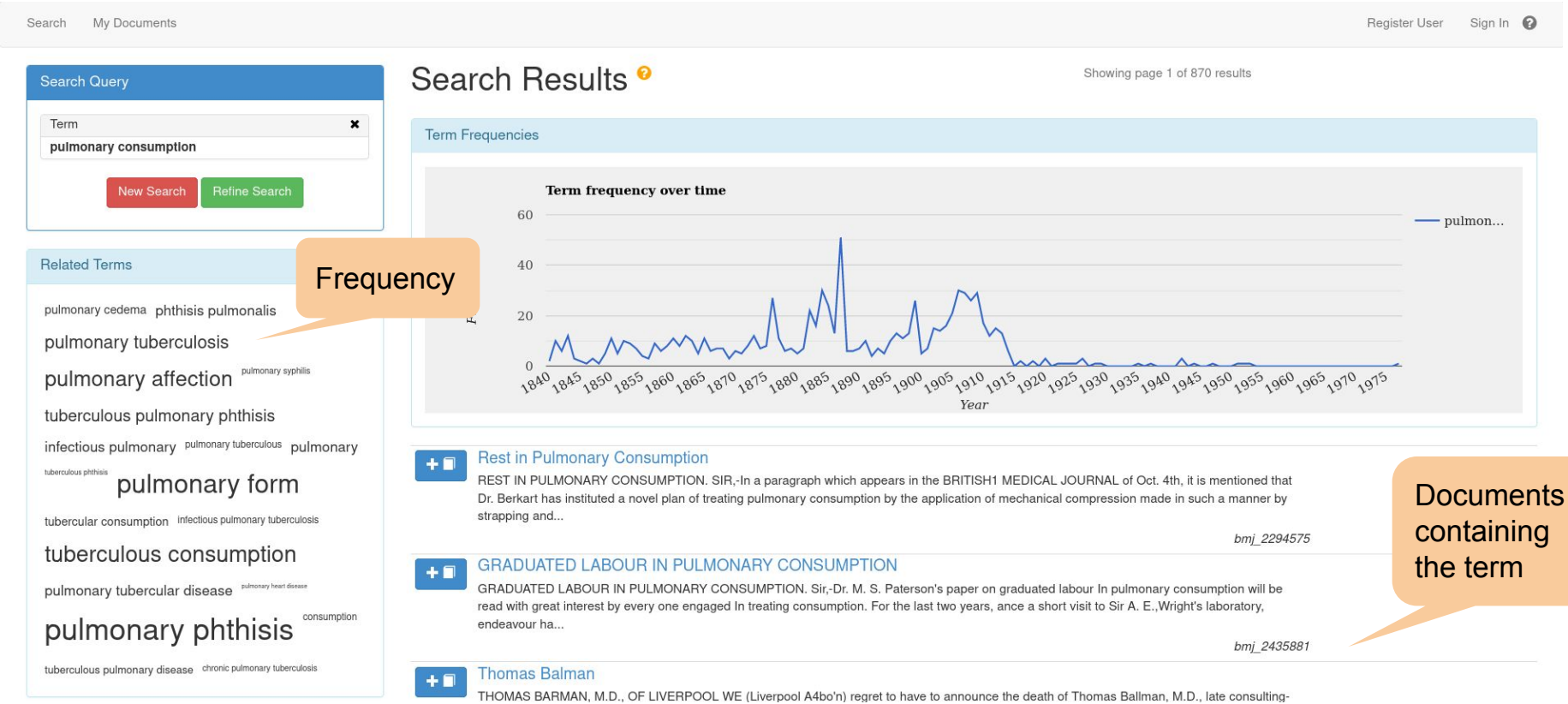


History of Medicine: Search Events





History of Medicine: Tracking



History of Medicine: Tracking

Search My Documents Register User Sign In

Search Query

Term ✖
pulmonary consumption

Term ✖
pulmonary tuberculosis

New Search | Refine Search

Related Terms

pulmonary y tuberculosis

tuberculous pulmonary disease · pulmonary cedema

bilateral pulmonary tuberculosis

miliary pulmonary tuberculosis

phthisis pulmonalis

nonpulmonary tuberculosis

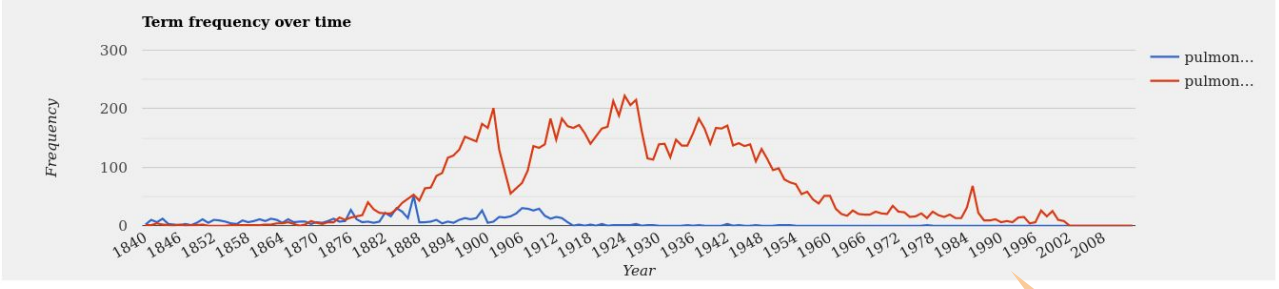
extrapulmonary tuberculosis

inactive pulmonary tuberculosis

Search Results

Showing page 1 of 11460 results

Term Frequencies



The Naval Manœuvres: Danger of Dissemination of Consumption

THE NAVAL MAN(3EUVRES: DANGER OF DIS SEMINATION OF CONSUMPTION. Sir,-The time for mobilisation of the fleet for the naval manceuvres being at hand, it may not be amiss to draw attention to the necessity for careful examination of the men, with a view to prevent any with pulmonary tuberculosis be...

bmj_2403738

PREVENTION OF PULMONARY TUBERCULOSIS

PREVENTION OF PULMONARY TUBERCULOSIS. A CONFERENCE of representatives of twenty-eight out of thee twenty-nine city and metropolitan borough councils was held on June 6th at the Paddington Town Hall to consider the question of taking measures to limit the spread of pulmonary tuberculosis. The chai...

bmj_2357676

Scotland

[FRO3 OUR SPECIAL CORRESPONDENTS.] GLASGOW OBSTETRICAL AND GYNAECOLOGICAL SOCIETY. ON Friday, June 28th, Professor Whitridge Williams, of Baltimore, delivered an address as Honorary President of the society. Taking as his subject the pernicious vomiting of pregnancy, he grouped the cases into tw...

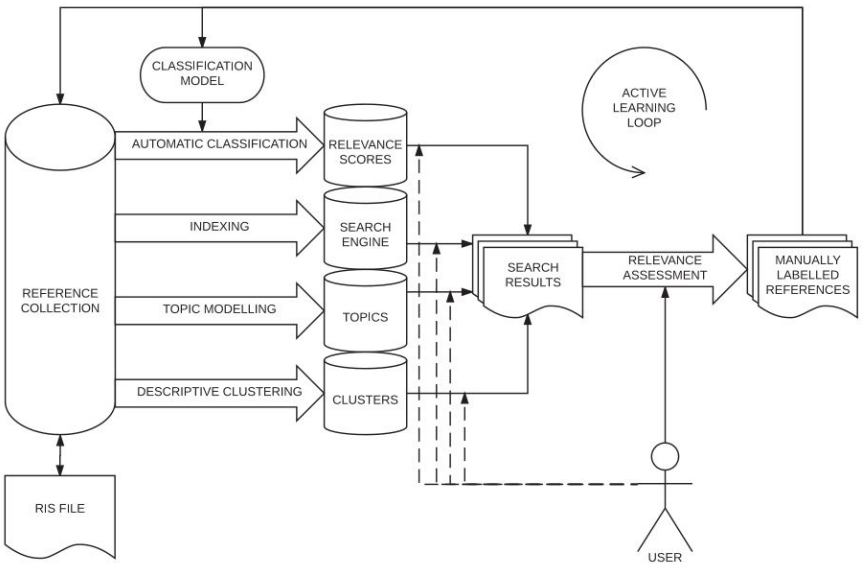
bmj_2333945

The second term is more frequently used after 1876

RobotAnalyst

Systematic review helper:

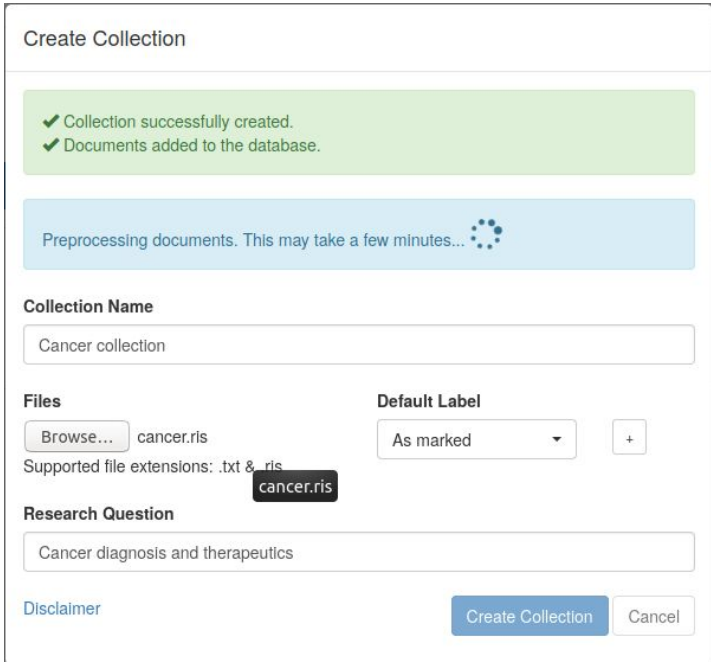
- Support for multiple collections
  - Easily updatable - tuned
- Multi-faceted & multi-level search
  - Quick search and integration of new publications
  - Search within a collection
  - Search within a cluster or topic
  - Faceted search
- Facilitated screening
  - Active learning
- Faceted search
- Topic analysis
- Automated clustering



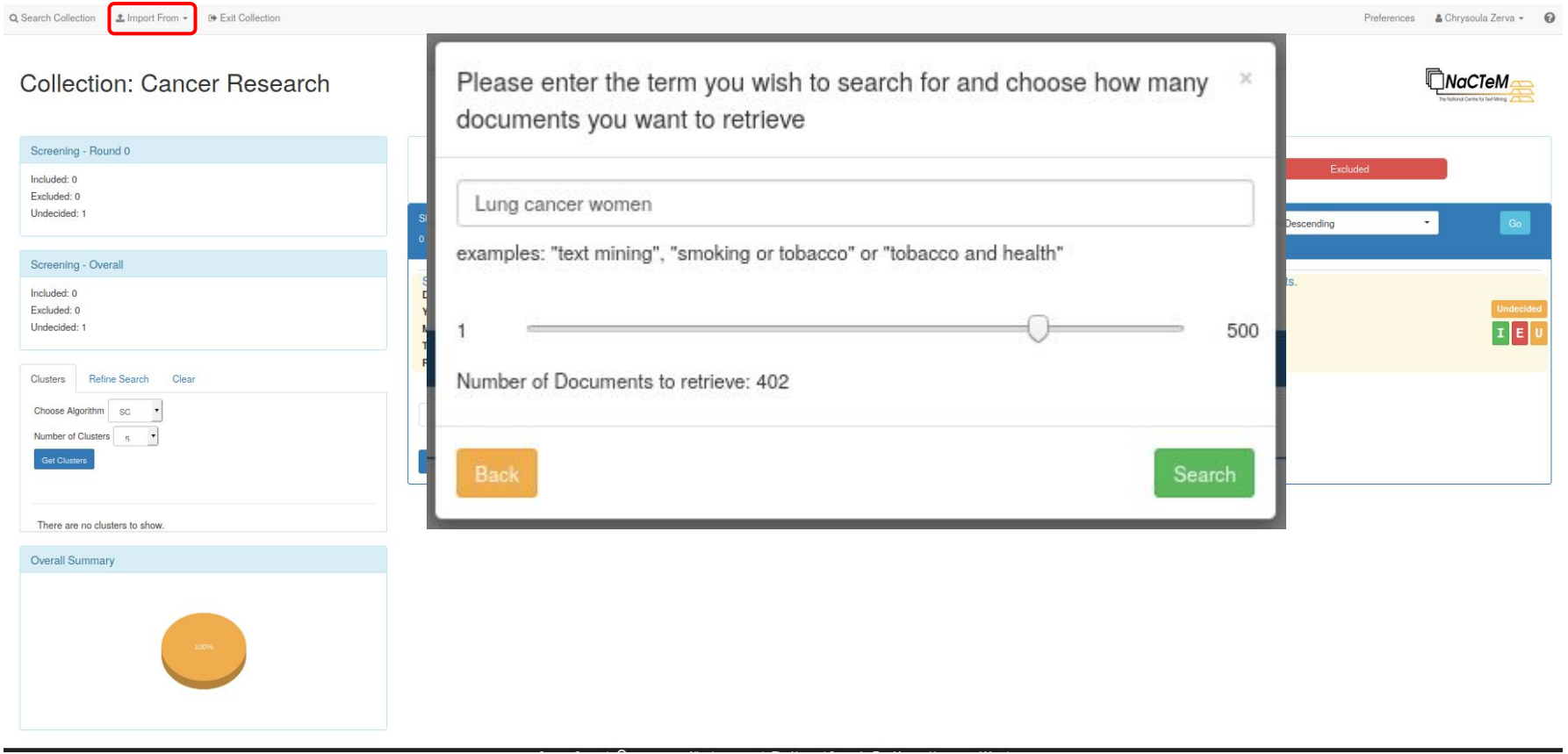
RobotAnalyst

Upload a set of initial references (or leave empty)

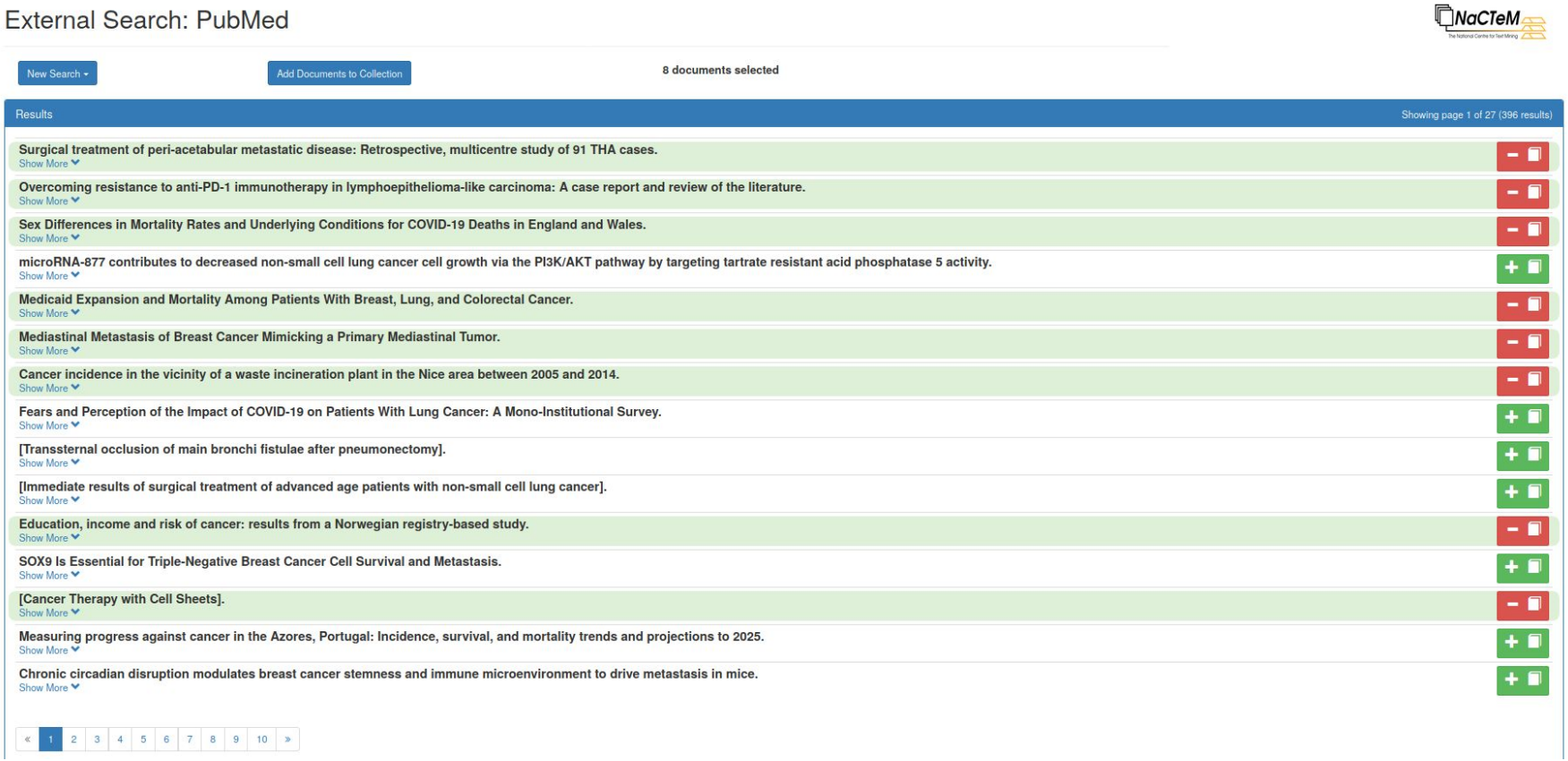
Specify main research question



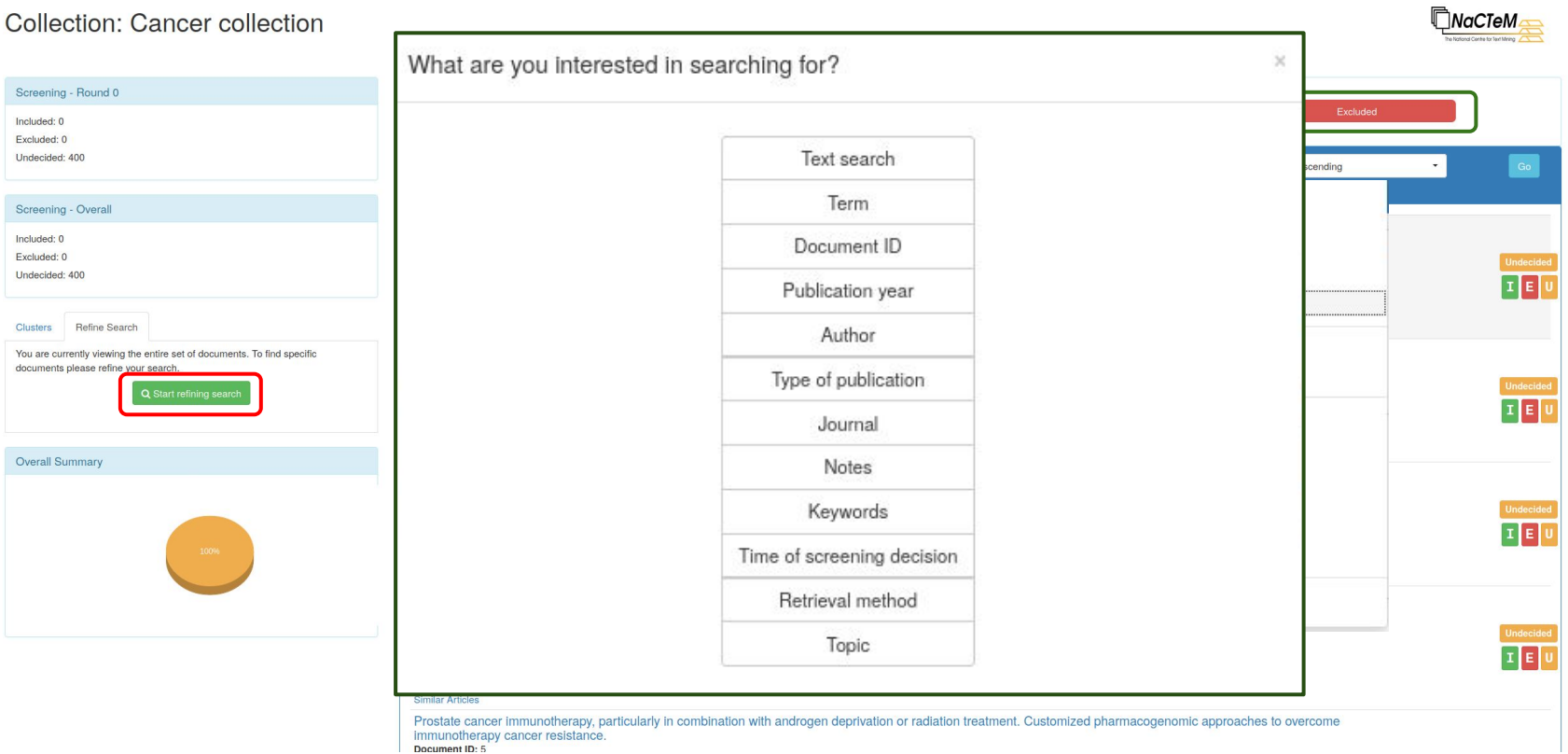
RobotAnalyst: Populate



RobotAnalyst: Populate

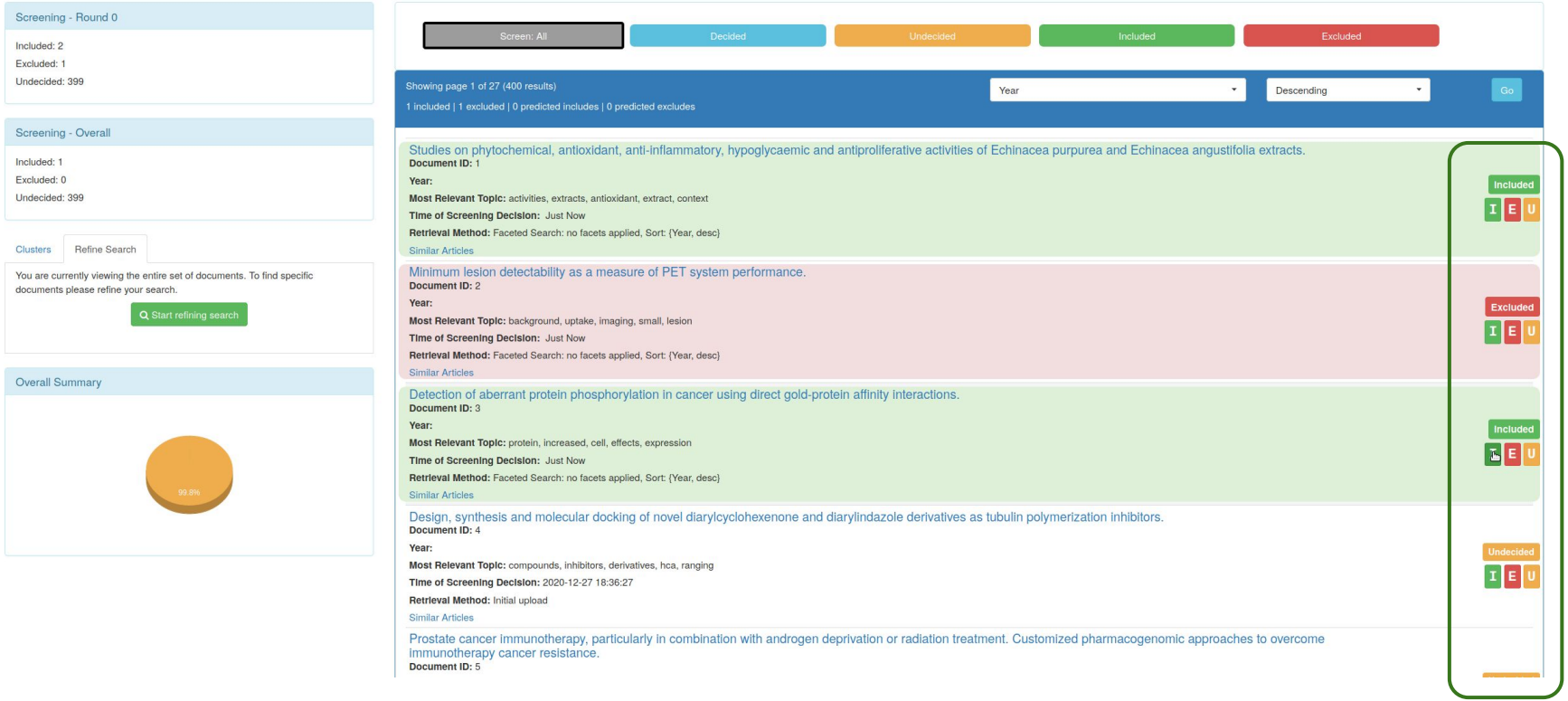


RobotAnalyst: Panel Overview

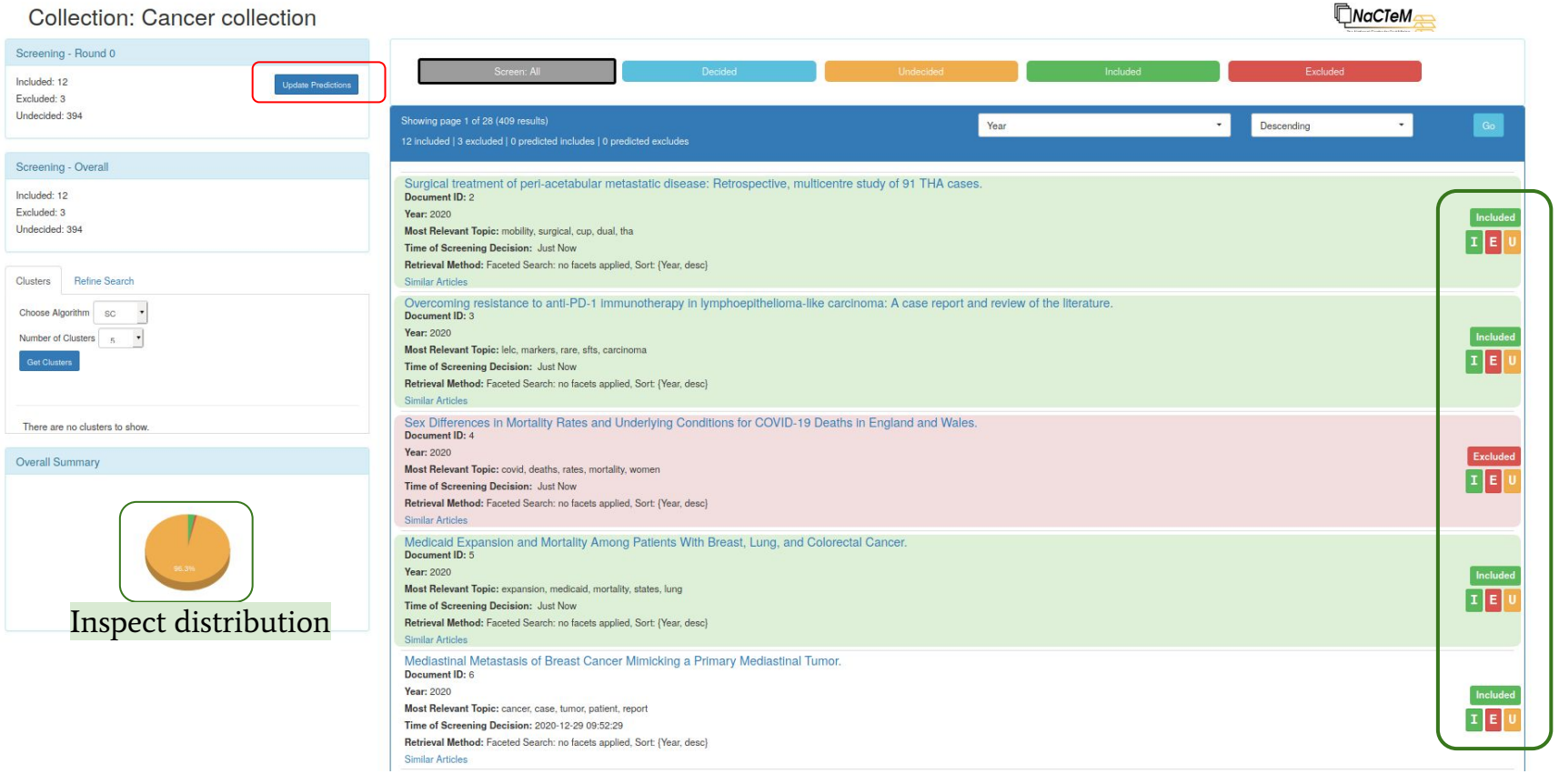


RobotAnalyst: Reviewing

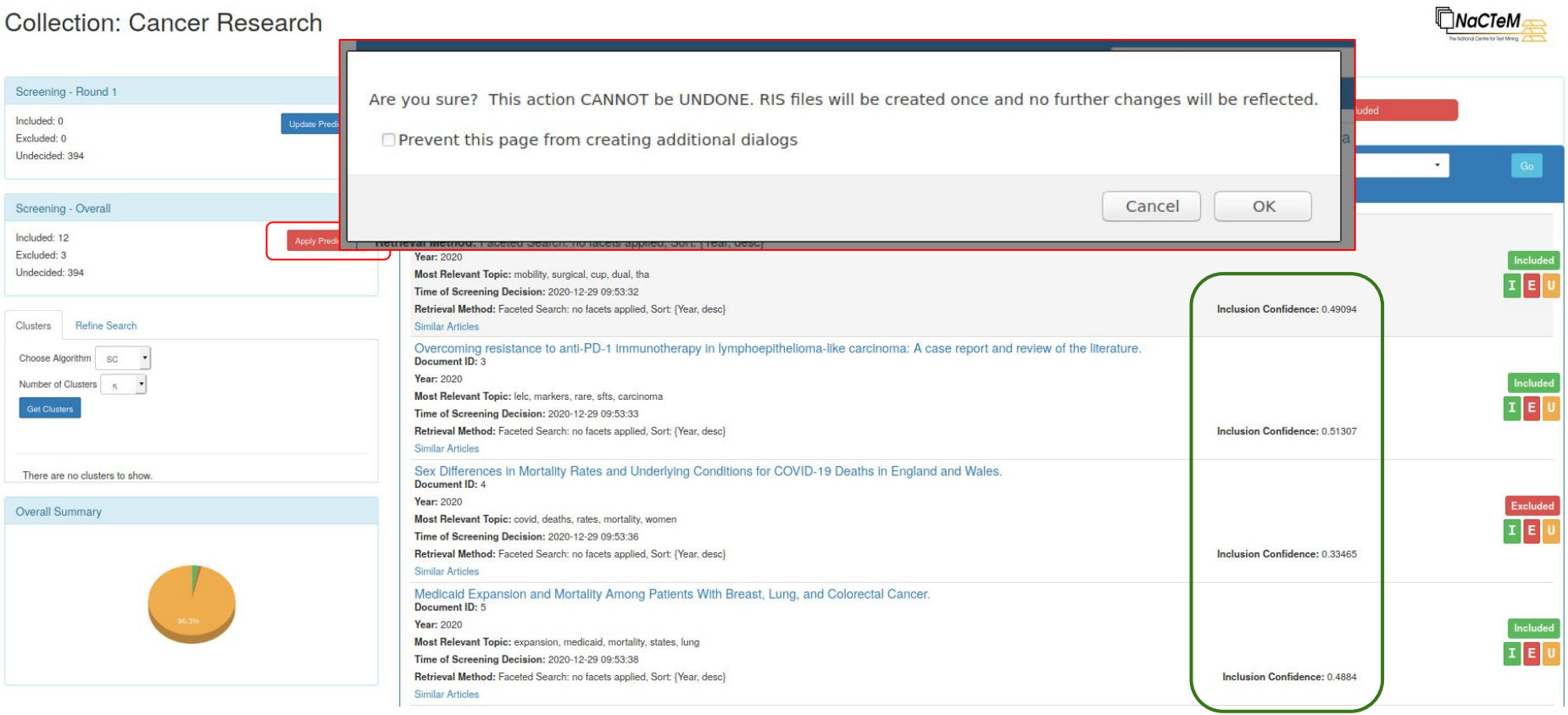
Collection: Cancer collection



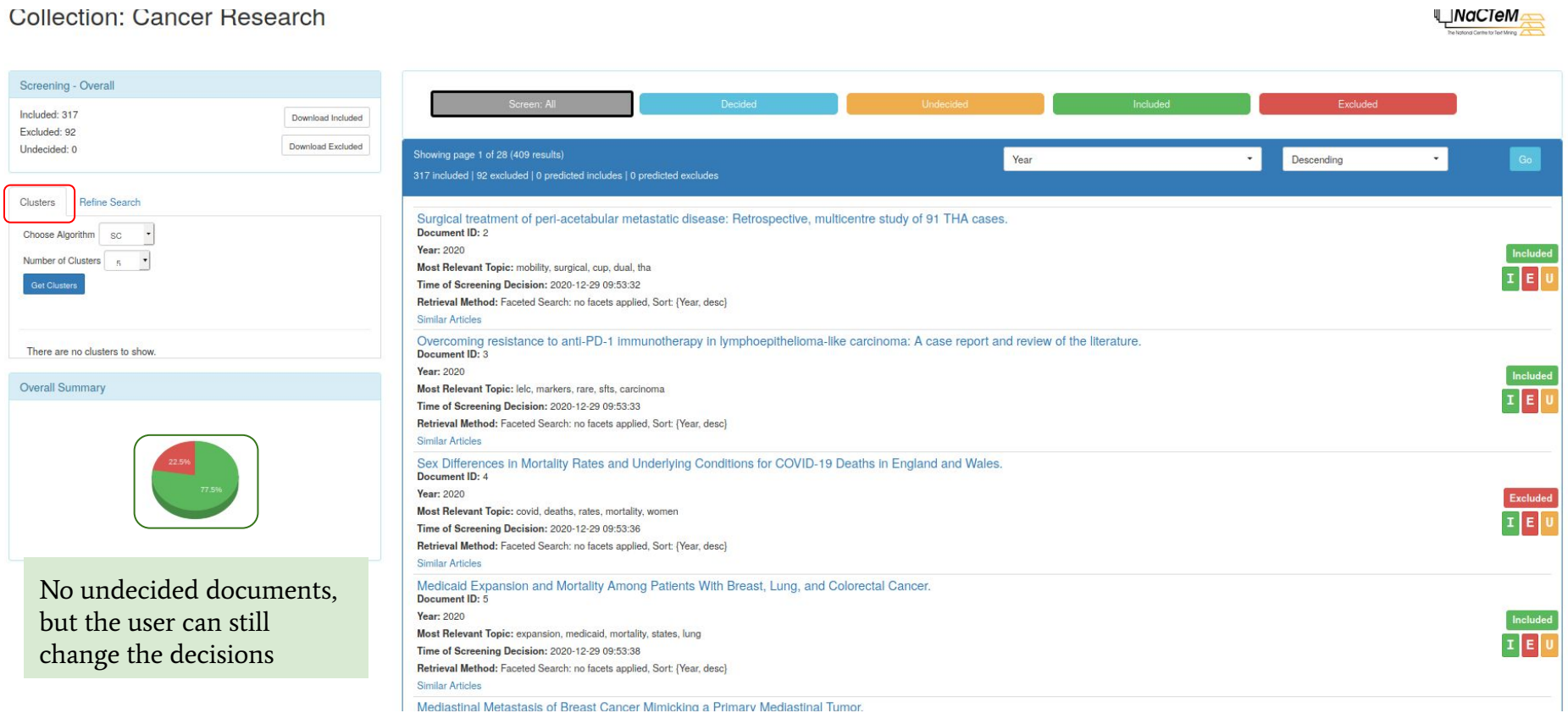
RobotAnalyst: Screening



RobotAnalyst: Automated screening



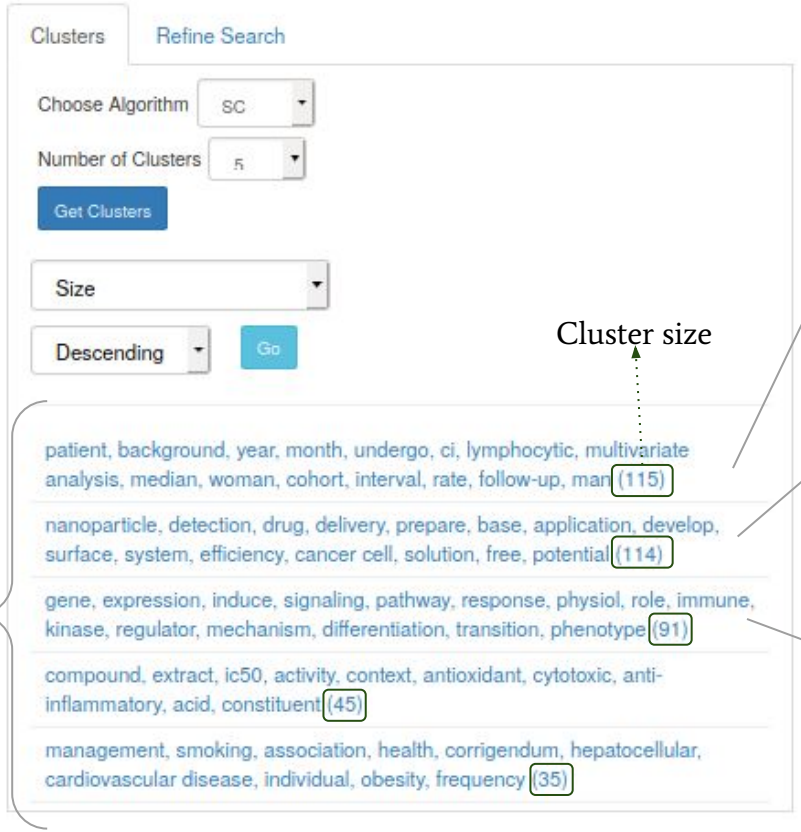
RobotAnalyst: Automated screening



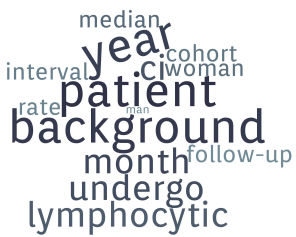
No undecided documents, but the user can still change the decisions

RobotAnalyst: Clustering

Username: efstathiachristopoulou
Password: thinarrangement57



Can be used for topic-based filtering bown (on click)

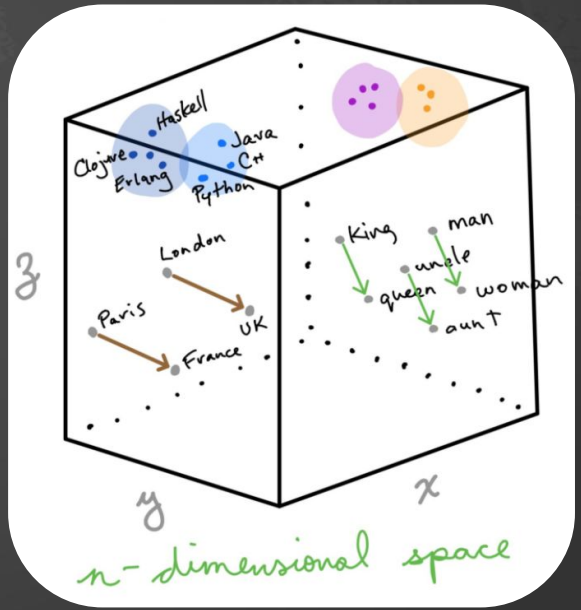


patient groups

drug discovery

gene expression

Representations



Source: Jon Krohn untapt Safari Live Lessons

Representations

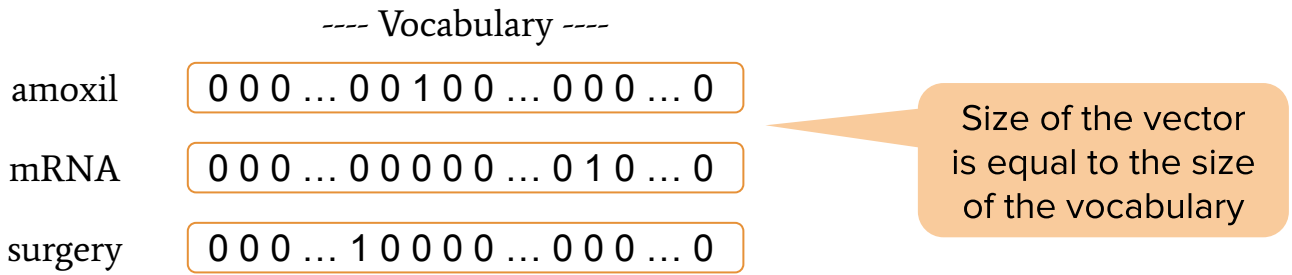
... aka, how can we encode language in a machine-readable format? 🤔

- The straightforward way: **Rules**
  - Cumbersome
  - Fail to adapt to the complexity of language
  - Not as expressive as natural language can be

- The more elegant-flexible-intuitive way: **Embeddings**
  - Useful properties - closer to natural language
  - Implicit encoding of rules - easily generalisable
  - Demonstrably state-of-the-art performance on NLP tasks

Language Models: Evolution

- Initial representations were discrete symbols, aka **One-hot vectors**
  - 1 in position i indicates the i-th word, rest positions are 0



Language Models: Evolution

- Initial representations were discrete symbols, aka **One-hot vectors**
  - 1 in position i indicates the i-th word, rest positions are 0
  - Drawbacks: Requirement for VERY large vocabularies
    No knowledge about some form of word “meaning”

amoxil	0 0 0 … 0 0 1 0 0 … 0 0 0 … 0
mRNA	0 0 0 … 0 0 0 0 0 … 0 1 0 … 0
surgery	0 0 0 … 1 0 0 0 0 … 0 0 0 … 0

Size of the vector is equal to the size of the vocabulary

Language Models: Evolution

- Initial representations were discrete symbols, aka **One-hot vectors**
  - 1 in position i indicates the i-th word, rest positions are 0
  - Drawbacks: Requirement for VERY large vocabularies
    No knowledge about some form of word “meaning”
- Rely on context to understand “meaning”, aka **Distributional Semantics**

> Words which frequently appear in similar contexts have similar meaning (Harris, 1954)

Language Models: Evolution

- Initial representations were discrete symbols, aka **One-hot vectors**
  - 1 in position i indicates the i-th word, rest positions are 0
  - Drawbacks: Requirement for VERY large vocabularies
    No knowledge about some form of word “meaning”
- Rely on context to understand “meaning”, aka **Distributional Semantics**

> Words which frequently appear in similar contexts have similar meaning (Harris, 1954)

  - ***Count-based methods***: Co-occurrence, Pointwise Mutual Information (PMI)
    ➜ Encode information based on corpus statistics
  - ***Prediction-based methods***: Word2Vec (Mikolov et al., 2013)
    ➜ Learn to *predict* the context of a word (or the word from its context)
  - ***Hybrid***: GloVe (Pennington et al., 2014)

Count-based Methods

Count-based methods construct a **word-context matrix** with certain elements

- Co-occurrence (Lund and Burgess, 1996)
  - ***Words***: words
  - ***Context***: A set of words in a fixed w-sized window (e.g. w = 3)
  - ***Matrix elements***: Number of times word w appears in context c

EGTA inhibited down-regulation of alpha-AR mRNA by NE .

	w_1	of	w_3	EGTA	w_5	w_6	NE	w_8	w_9	
alpha-AR	0	1	0	1	0	0	0	0	0	...

- Typically, high co-occurrence ➔ high similarity

Count-based Methods

Count-based methods construct a **word-context matrix** with certain elements

- Pointwise Mutual Information
  - ***Words***: words
  - ***Context***: A set of words in a fixed w-sized window (e.g. w = 3)
  - ***Matrix elements***: PMI - a measure of association

Positive pmi indicates that words are correlated above chance!

$$pmi(x;y) = \log \frac{p(x,y)}{p(x)p(y)}$$

Probability words x and y co-occur

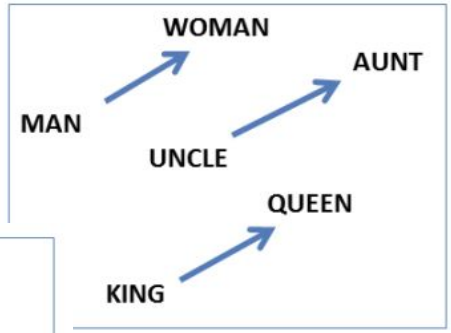
Probability of word x and word y

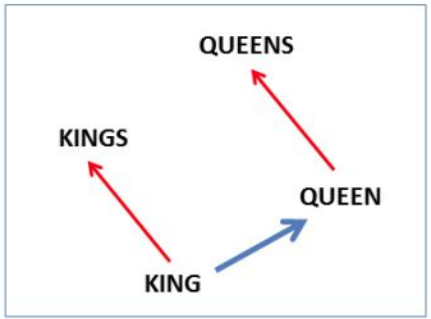
$$ppmi(x;y) = max(0, pmi(x;y))$$

	w_1	of	w_3	EGTA	w_5	w_6	NE	w_8	w_9	
alpha-AR	0	1.22	0	-3.78	0	0	0	0	0	...

Prediction-based methods: Word2Vec

- Word embeddings are *learned* instead of constructed manually
  - ***Words***: words
  - ***Context***: A set of words in a fixed w-sized window (e.g. w = 3)
  - ***Probabilities***: Computed for each target word for each of their context
- Advantages
  - Fast training
  - High quality word embeddings
  - Learning embeddings have some nice properties
  - Small dimensionality !





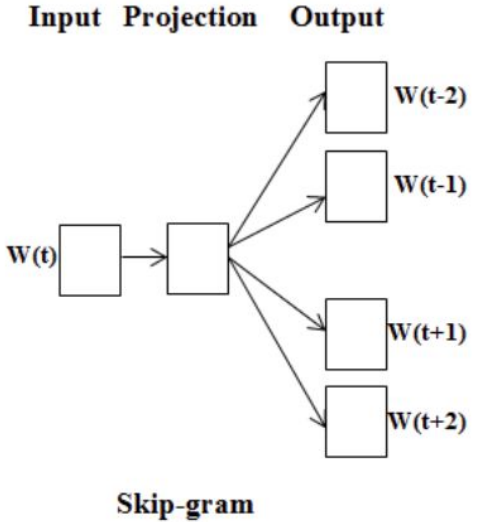
Algorithms:

1. *Skip-Gram*: Predict the context given a word
2. *CBoW*: Predict a word given the context

Prediction-based methods: Word2Vec

Skip-gram

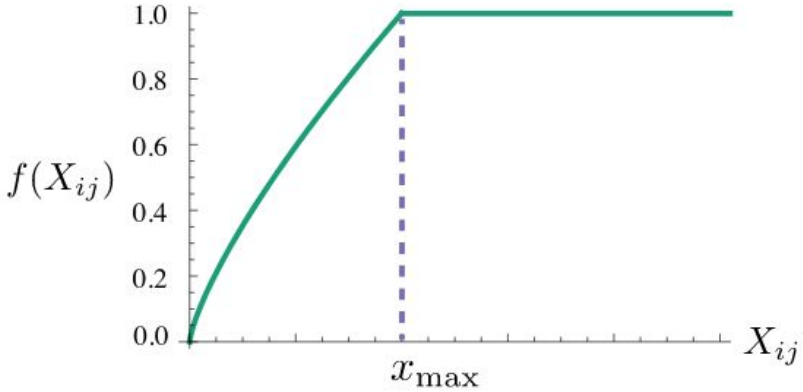
- Most famous and commonly used method
- Training is done in pairs using a **sliding window** *w*
- Learnable embeddings via gradient descent
- 2 vectors are produced for each word
  - 1 as target word, 1 as context word
- **Negative Sampling** makes things faster
  - Use a subset of K less frequent words as context



Sentence	Training Samples
EGTA inhibited down-regulation of alpha-AR mRNA	(EGTA, inhibited) (EGTA, down-regulation)
EGTA inhibited down-regulation of alpha-AR mRNA	(inhibited, EGTA) (inhibited, down-regulation) (inhibited, of)
EGTA inhibited down-regulation of alpha-AR mRNA	(down-regulation, EGTA) (down-regulation, inhibited) (down-regulation, of) (down-regulation, alpha-AR)

Hybrid: GloVe

- **Combination** of **count-based** and **prediction-based** methods
- GloVe = **Glo**bal **Ve**ctors
- Learned vectors via gradient descent



Target word vector

Context word vector

$$J = \sum_{i,j=1}^{V} f\left(X_{ij}\right)\left(w_i^T \tilde{w}_j + b_i + \tilde{b}_j - \log X_{ij}\right)^2$$

Co-occurrence frequency

$$f(x) = \begin{cases} (x/x_{\max})^{\alpha} & \text{if } x < x_{\max} \\ 1 & \text{otherwise} \end{cases}$$

Weighting function

The weighting function
- Penalises rare events
- Not overweight frequent events

Language Models: Evolution

- Initial representations were discrete symbols, aka **One-hot vectors**
  - 1 in position i indicates the i-th word, rest positions are 0
  - Drawbacks: Requirement for VERY large vocabularies
    No knowledge about some form of word "meaning"
- Rely on context to understand "meaning", aka **Distributional Semantics**

> Words which frequently appear in similar contexts have similar meaning (Harris, 1954)

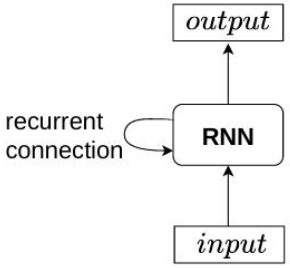
  - ***Count-based methods***: Co-occurrence, Pointwise Mutual Information (PMI)
    ➜ Encode information based on corpus statistics
  - ***Prediction-based methods***: Word2Vec (Mikolov et al., 2013)
    ➜ Learn to *predict* the context of a word (or the word from its context)
  - ***Hybrid***: GloVe (Pennington et al., 2014)
    Drawbacks: The encoded "meaning" remains the same across different contexts

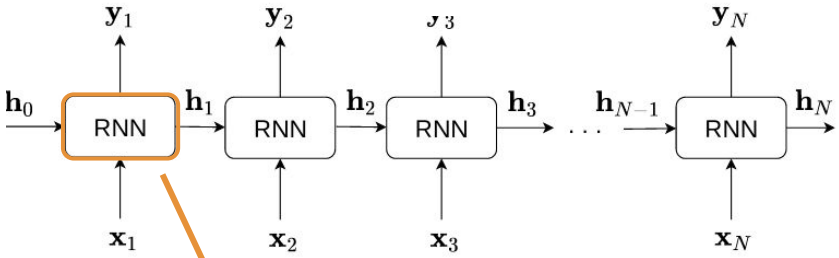
Language Models: Evolution

- Different embeddings for different contexts, aka **Contextual Word Embeddings**
  - **ELMo** which is based on Bi-LSTMs (Peters et al., 2018)
  - Predict the next word in a sentence given the current (and previous) words

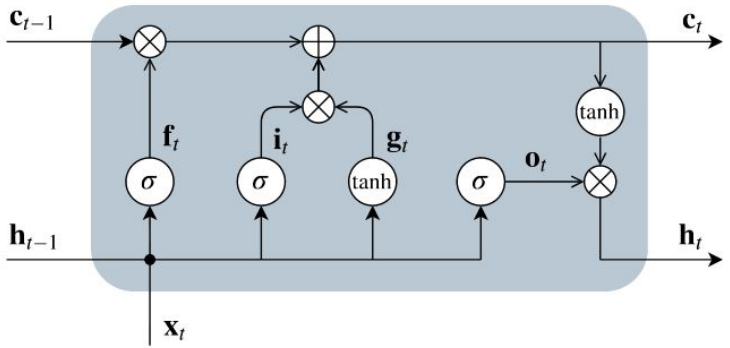
Recap: RNNs & LSTMs

- **R**ecurrent **N**eural **N**etworks (Elman, 1990)
- Previous elements used as input
- Have an internal “memory” aka hidden state (**h**)
- Contextual word embeddings (**y**)
- Drawback: Cannot model very long dependencies



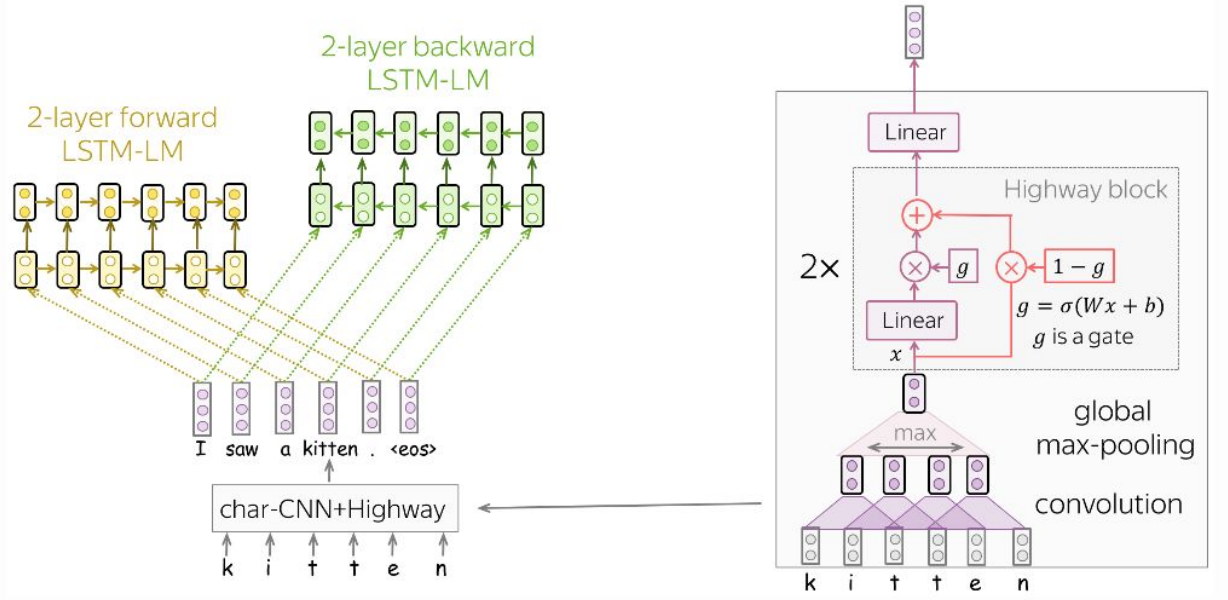


- **L**ong-**S**hort **T**erm **M**emory (Hochreiter et al., 1997)
- Able to learn long dependencies
- The LSTM cell has 4 layers called “gates”
- LSTM cell is our “memory”
  - *Forget gate*: How much information we want to forget
  - *Input gate*: Amount of incoming information
  - *Output gate*: Amount of memory revealed to next state
  - *G gate*: Simple RNN cell



ELMo: Embeddings for Language Models

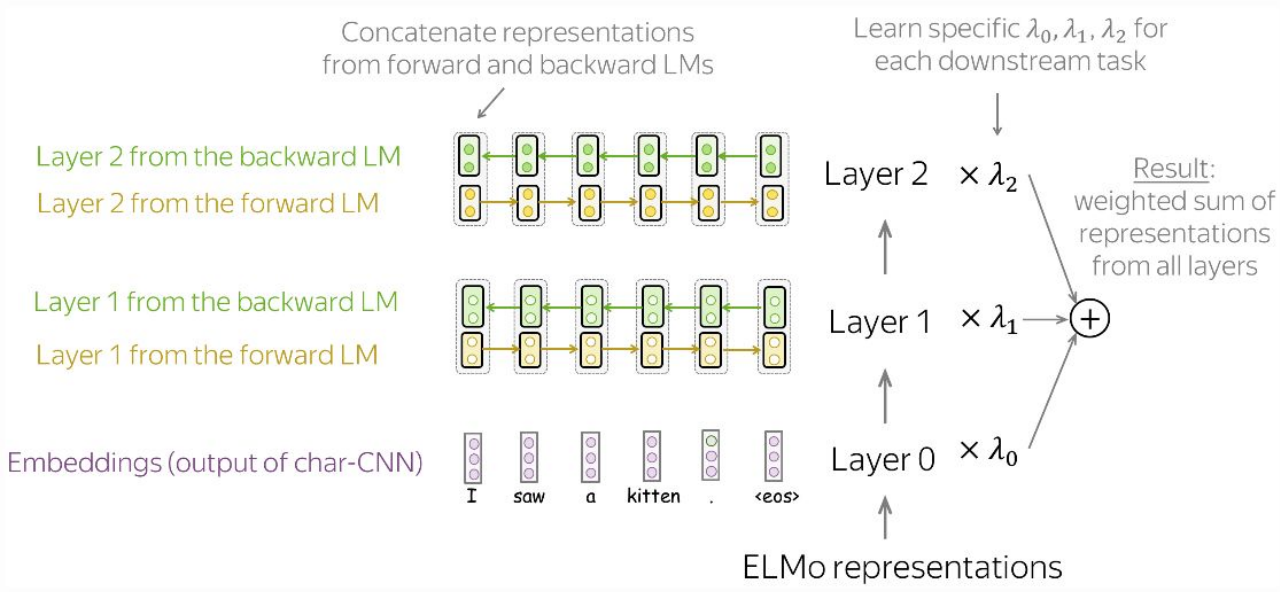
- Produce contextualised word embeddings by looking at the entire sentence
- Different context - different embedding
- Successor of the CoVe model (McCann et al., 2017) introduced for Machine Translation
- Able to fine-tune it on downstream tasks !



* Credits to: https://lena-voita.github.io/nlp_course/transfer_learning.html

ELMo: Embeddings for Language Models

- Produce contextualised word embeddings by looking at the entire sentence
- Different context - different embedding
- Successor of the CoVe model (McCann et al., 2017) introduced for Machine Translation
- Able to fine-tune it on downstream tasks !



* Credits to: https://lena-voita.github.io/nlp_course/transfer_learning.html

Language Models: Evolution

- Different embeddings for different contexts, aka **Contextual Word Embeddings**
  - **ELMo** which is based on Bi-LSTMs (Peters et al., 2018)
  - Predict the next word in a sentence given the current (and previous) words
  - Drawbacks: Very *slow* to train
    Rely mostly on word order making it easy to “forget” long dependencies

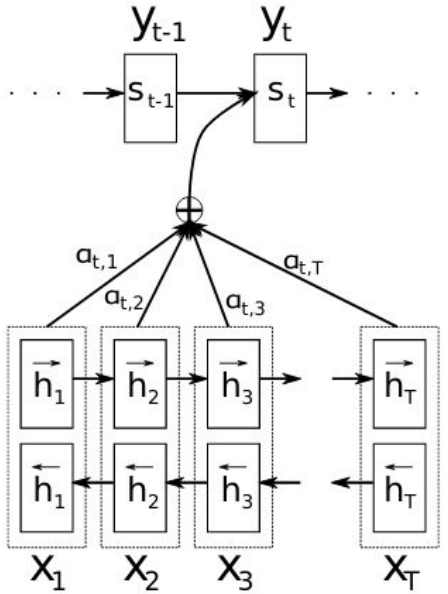
Language Models: Evolution

- Different embeddings for different contexts, aka **Contextual Word Embeddings**
  - **ELMo** which is based on Bi-LSTMs (Peters et al., 2018)
  - Predict the next word in a sentence given the current (and previous) words
  - Drawbacks: Very *slow* to train
    Rely mostly on word order making it easy to “forget” long dependencies

- From word order to attention, aka **Transformers**
  - ***Attention*** heads learn which words are important when looking at a certain word
  - ***Masked Language Modeling***: Fill-in the blank (i.e. find the word given the context)

Attention in Neural Networks

- Concept originally adapted from computer vision
  - Learning to “attend” to specific places in the image
- Measures the “importance” of a representation towards another
- Utilised first in NLP for machine translation ([Bahdanau et al., 2015](#))
- Attention comes in many flavours
  - Additive, Dot-product, …



$$\mathbf{c}_t = \sum_{i=1}^{n} \alpha_{t,i} \boldsymbol{h}_i \quad ; \text{Context vector for output } y_t$$

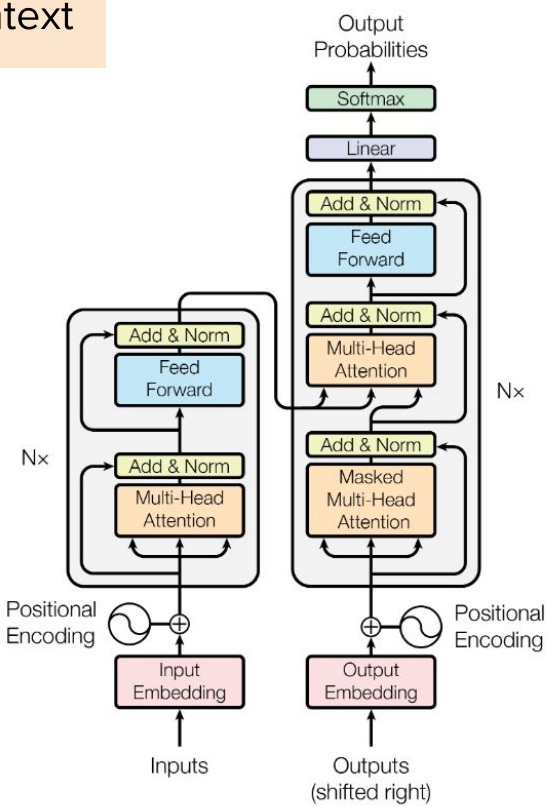
$$\alpha_{t,i} = \text{align}(y_t, x_i) \quad ; \text{How well two words } y_t \text{ and } x_i \text{ are aligned.}$$

$$= \frac{\exp(\text{score}(\boldsymbol{s}_{t-1}, \boldsymbol{h}_i))}{\sum_{i'=1}^{n} \exp(\text{score}(\boldsymbol{s}_{t-1}, \boldsymbol{h}_{i'}))} \quad ; \text{Softmax of some predefined alignment score..}$$

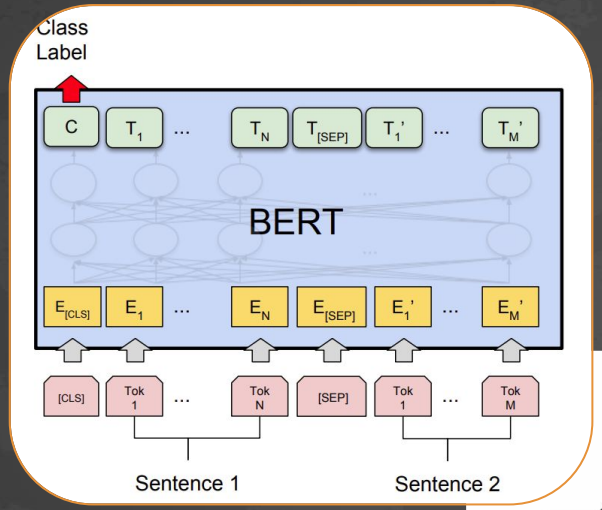
Transformers

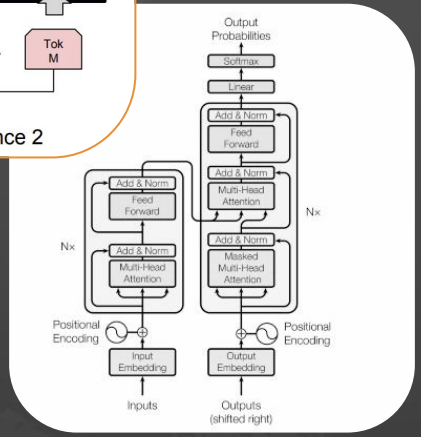
Use of **attention** to capture word context

- Introduced in Vaswani et al. (2018)
- Use of attention-heads to identify significant words for each concept representation
- Stack multiple transformers blocks to capture more distant dependencies
- Training: Masked Language Modelling
  - GPT (Radford et al., 2019): next token prediction
  - BERT (Devlin et al., 2018): next token prediction + next sentence prediction
  - ELECTRA (Clark et al., 2020): Valid/Invalid token prediction



Architectures





Positional Features

- Such features are mostly common in *Relation Extraction*
- They are used to encode information about the “place” of a word in a snippet

1. **Entity Markers** (Zhang et al., 2015; Soares et al., 2019)
   - Explicitly learn representations for them as additional words

<e1>EGTA</e1> inhibited down-regulation of <e2>alpha-AR</e2> mRNA by NE

2. **Relative Positions** (Collobert et al., 2011; Zeng et al., 2014, Zhang et al., 2017)
   - Measure the relative position of one word towards another

3. **Position Indicators** (Soares et al., 2019)
   - indicator embedding layer

EGTA inhibited down - regulation of alpha - AR mRNA by NE
1 0 0 0 0 0 2 2 2 0 0 0

Linguistic features

- Linguistic features are designed to help models encode linguistic information into their vector representations

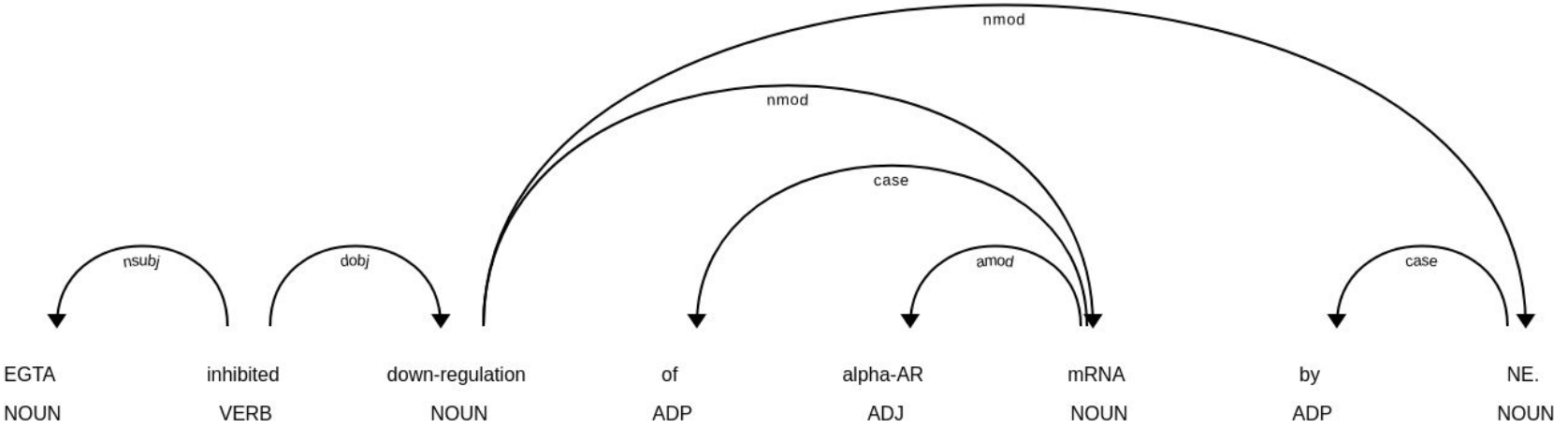
- Most commonly used Linguistic features:
  - **Part-of-speech** (POS) **tags**: VERB, NOUN, etc

EGTA	inhibited	down-regulation	of	alpha-AR	mRNA	by	NE
NOUN	*VERB*	*NOUN*	*ADP*	*ADJ*	*NOUN*	*ADP*	*NOUN*

Linguistic features

- Linguistic features are designed to help models encode linguistic information into their vector representations

- Most commonly used Linguistic features:
  - **Part-of-speech** (POS) **tags**: VERB, NOUN, etc
  - **Dependency paths** (most common to use the shortest dependency path: SDP)



Linguistic features

- Linguistic features are designed to help models encode linguistic information into their vector representations

- Most commonly used Linguistic features:
  - **Part-of-speech** (POS) **tags**: VERB, NOUN, etc
  - **Dependency paths** (most common to use the shortest dependency path: SDP)
  - **Lemmas**: Assign the basic form of a word, e.g. “was” ➔ “be”, “breathing” ➔ “breathe”

EGTA	inhibited	down-regulation	of	alpha-AR	mRNA	by	NE
EGTA	*inhibit*	down-regulation	*of*	alpha-AR	*mRNA*	*by*	*NE*

Linguistic features

- Linguistic features are designed to help models encode linguistic information into their vector representations

- Most commonly used Linguistic features:
  - **Part-of-speech** (POS) **tags**: VERB, NOUN, etc
  - **Dependency paths** (most common to use the shortest dependency path: SDP)
  - **Lemmas**: Assign the basic form of a word, e.g. “was” ➔ “be”, “breathing” ➔ “breathe”
  - **Surface Form**: The form in which a word appears
  - **Characters**: The characters that comprise a word
  - **Word Length**
  - **Semantics**: Different meanings of a word, e.g. WordNet synsets

Synsets	*Definition*
suppress	to put down by force or authority
inhibit	limit the range or extent of
inhibit	limit, block, or decrease the action or function of
inhibit	control and refrain from showing; of emotions, desires, impulses, or behavior

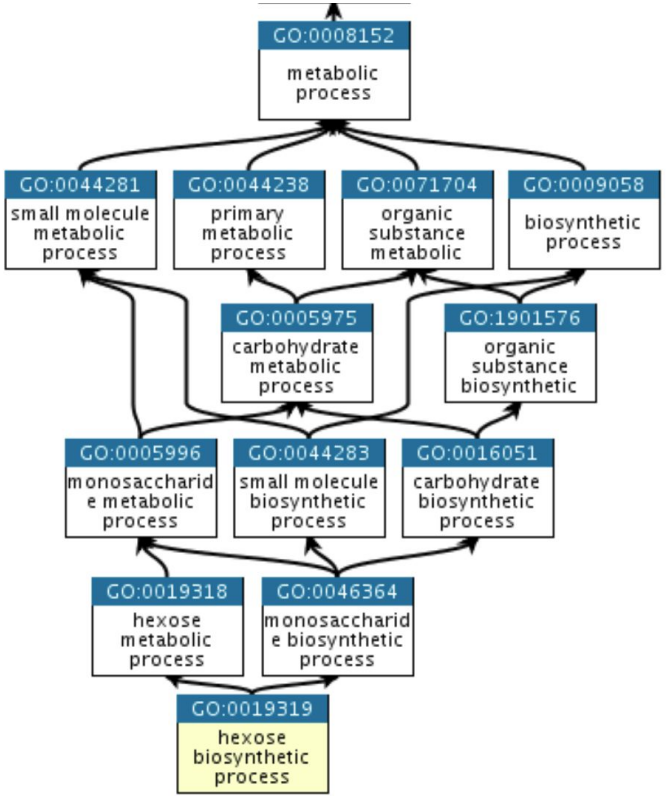
External Features

- External features can be information derived from:
  - ***Dictionaries***:
    - ➜ Included in the dictionary (1), not included (0)

Dictionary Category	Examples
Causation	because, effect
Articles	a, an, the
Negation	no, not, never
Exclusives	but, without, exclude
Nonfluencies	er, hm, umm
Positive emotion	love, nice, sweet
Negative emotion	hurt, ugly, nasty

External Features

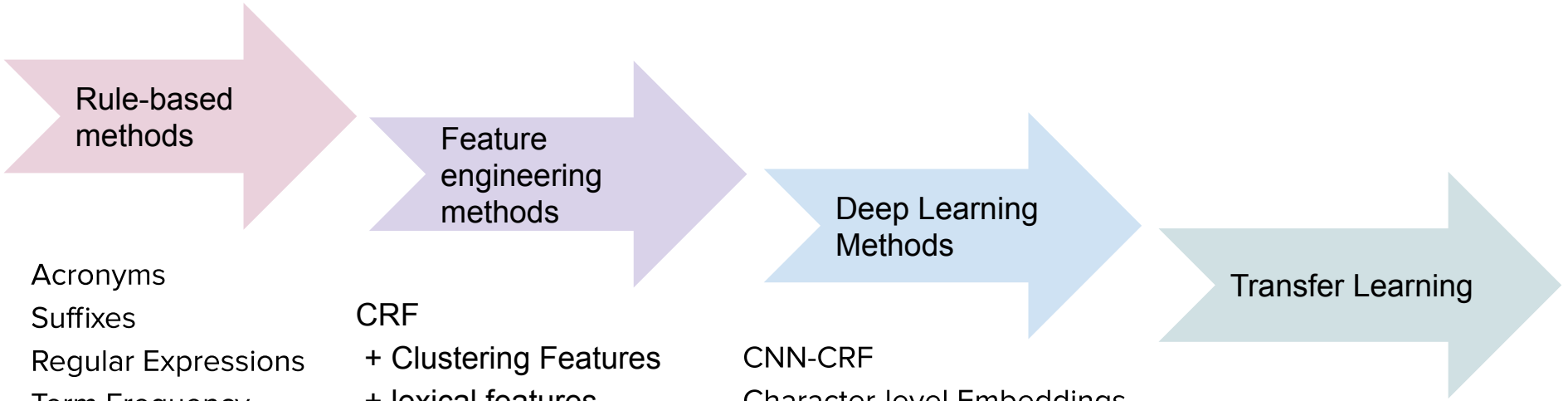
- External features can be information derived from:
  - ***Dictionaries***:
    - ➜ Included in the dictionary (1)
  - ***Ontologies***:
    - ➜ N-step hypernyms of hyponyms of an enti



External Features

- External features can be information derived from:
  - ***Dictionaries***:
    - ➜ Included in the dictionary (1), not included (0)
  - ***Ontologies***:
    - ➜ N-step hypernyms of hyponyms of an entity
  - ***Knowledge Bases***: *Source of factual knowledge*
    - ➜ Associations with other objects in the KB
    - ➜ Semantic types if the entity exists in the KB

Named Entity Recognition



Named Entity Recognition

1. Initial approaches were **rule-based** (Hettne et al., 2009)
   Eg. *Detection of Acronyms*:
   words with caps only, more than one cap, letters and digits, etc

2. Later on, **Machine Learning** models were used with **feature engineering**
   - Tagging schemes, Gazetteers (Ratinov and Roth, 2009)
     - Early indication that pre-training on large resources will be useful!
   - CRF + Clustering features (Lin and Wu, 2009)
   - CRF + Phrase embeddings (Passos et al., 2014)

Conditional Random Field (CRF) (Lafferty et al., 2001)
- A probabilistic model that takes into account neighboring samples
- Handles dependencies between the samples
- Linear chain CRF is thus suitable for modelling the dependencies among sentential words

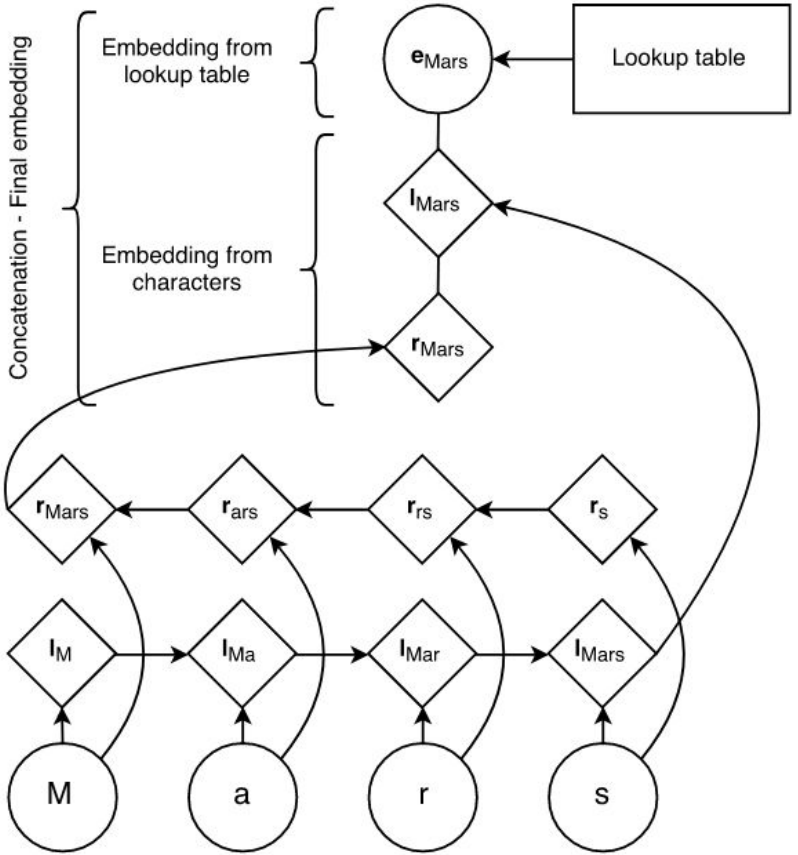
NER: Neural Approaches

➔ **BiLSTM-CRF-Char**
Proposed by [Lample et al. (2016)]

- [Nogueira and Guimaraes (2015)] proposed **character-level embeddings** based on CNN
- Useful for morphologically rich languages
- Helpful for out-of-vocabulary words

Steps for Character Embeddings:

1. Feed characters of a word into a BiLSTM
2. Concatenate the hidden states of forward and backward LSTM
3. Concatenate these with the word embedding



NER: Neural Approaches

Word Embeddings as input:

- Created as we saw earlier

BiLSTM Layer:

- Concatenation of forward and backward states → [h_forw ; h_back]

CRF Layer:

Transition scores matrix

sequence

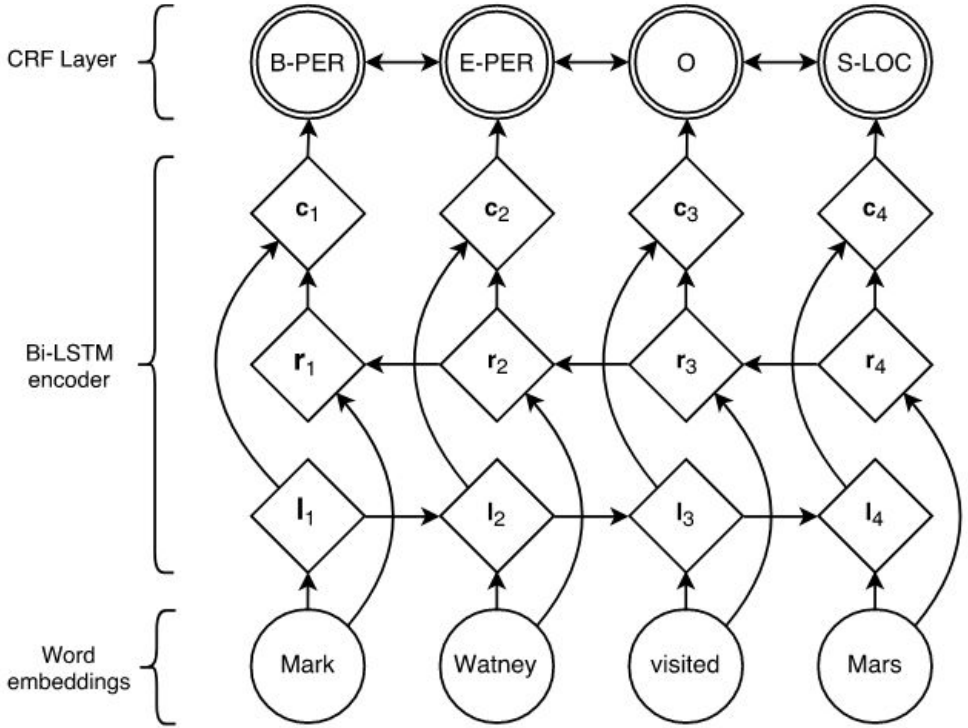
labels

logits

$$s(\mathbf{X}, \mathbf{y}) = \sum_{i=0}^{n} A_{y_i, y_{i+1}} + \sum_{i=1}^{n} P_{i, y_i}$$

$$\mathbf{y}^* = \underset{\tilde{\mathbf{y}} \in \mathbf{Y_X}}{\operatorname{argmax}}\, s(\mathbf{X}, \tilde{\mathbf{y}})$$

All possible tag sequences



NER: Neural Approaches

Subsequent approaches experimented with several combinations:

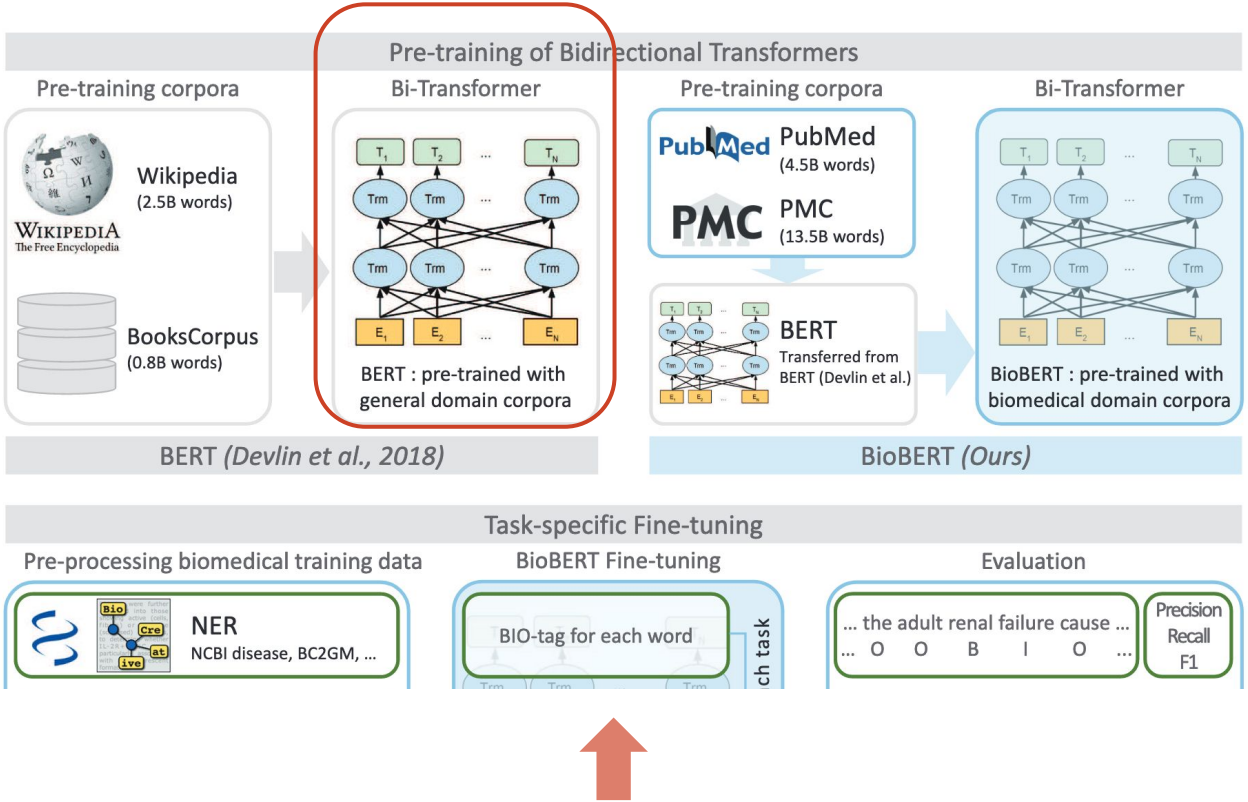
- RNN-Char (Sahu and Anand, 2016)
  - Comparison of multiple RNN architectures
  - Showed the **RNNs perform better than CNNs on the NER task**

- BiLSTM-CNN-Char (Chiu and Nichols, 2016; Kocaman and Talby, 2020)
  - **Convolutional layer on characters** is better that sequential!

- BiLSTM-CRF with dependencies (Jie and Lu, 2019)
  - Strong correlations between the entity types and dependency relations

- BiLSTM-CRF with ELMo (Dai et al., 2019)
  - **CRF** continues to be used as an **effective feature**
  - **Pre-training** on large corpora benefits representations and performance

NER: Pre-trained LMs



BioBERT (Lee et al., 2019)

- Initialised on BERT, trained on:
  - English Wikipedia (2.5B words)
  - BooksCorpus (0.8B words)
- Further trained on:
  - PubMed Abstracts (4.5B words)
  - PMC Full-text Articles (13.5B words)

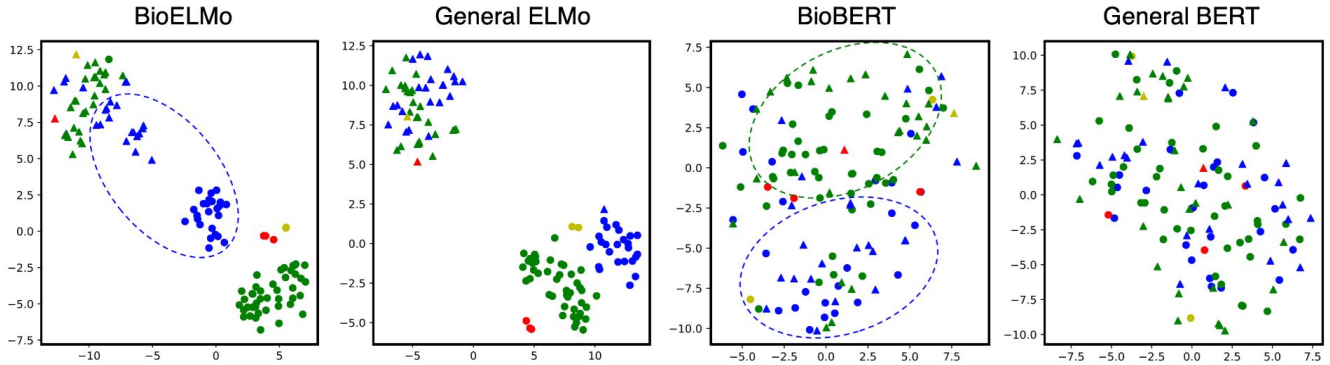
NER: Pre-trained LMs

Clinical BERT (Alsentzer et al., 2019) ➔ Trained on 2M nodes from MIMIC-III

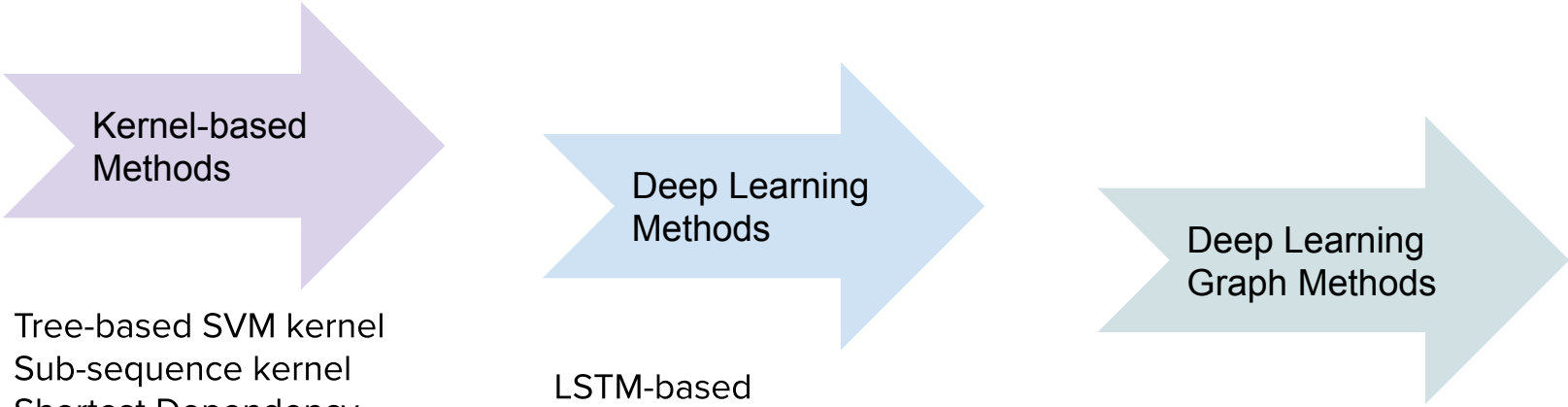
Model	i2b2 2006	i2b2 2010	i2b2 2012	i2b2 2014
BERT	93.9	83.5	75.9	92.8
BioBERT	**94.8**	86.5	78.9	**93.0**
Clinical BERT	91.5	86.4	78.5	92.6
Discharge Summary BERT	91.9	86.4	78.4	92.8
Bio+Clinical BERT	94.7	87.2	**78.9**	92.5
Bio+Discharge Summary BERT	94.8	**87.8**	78.9	92.7

In-domain training vs out-of-domain training

BioELMo
(Jin et al., 2019)
→ Trained on PubMed (10M recent abstracts, 2.46B tokens)



Relation Extraction



Tree-based SVM kernel
Sub-sequence kernel
Shortest Dependency Path

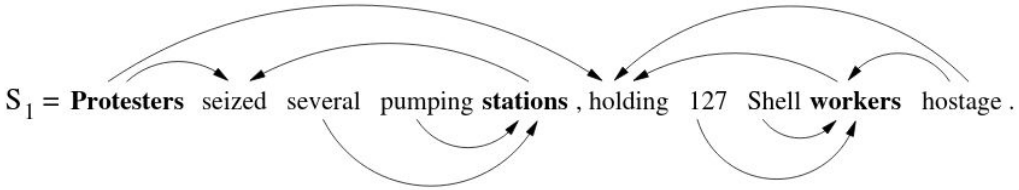
LSTM-based
CNN-based
- + Positional Features
- + Tree-structures
- + Attention

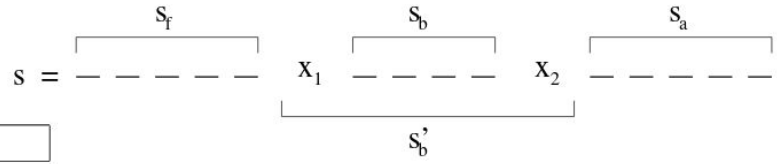
Pre-trained LMs
- ○ GPT-2
- ○ BERT

Relation Extraction

Kernel-based Methods

- Relying on using Support Vector Machines (SVMs) with different kernels
  - Sub-sequence Kernel (Bunescu and Mooney 2005)
  - Tree-kernel (Zelenco et al., 2003; Bunescu and Mooney, 2006)

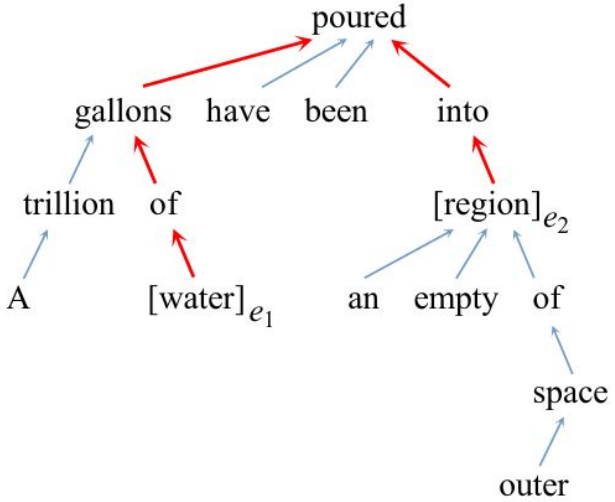




Relation Instance	Shortest Path in Undirected Dependency Graph
S_1: protesters AT stations	**protesters** ⟶ seized ⟵ **stations**
S_1: workers AT stations	**workers** ⟶ holding ⟵ protesters ⟶ seized ⟵ **stations**
S_2: troops AT churches	**troops** ⟶ raided ⟵ **churches**
S_2: ministers AT churches	**ministers** ⟶ warning ⟵ troops ⟶ raided ⟵ **churches**

Relation Extraction: Neural Approaches

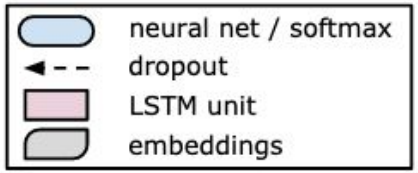
- Using **position features**
  - CNN + positions (Zeng et al., 2014)
  - RNN + max pooling + **entity markers** (Zhang and Wang, 2015)
- Using **tree-structures**
  - LSTM + Shortest Dependency Path (Xu et al., 2015)

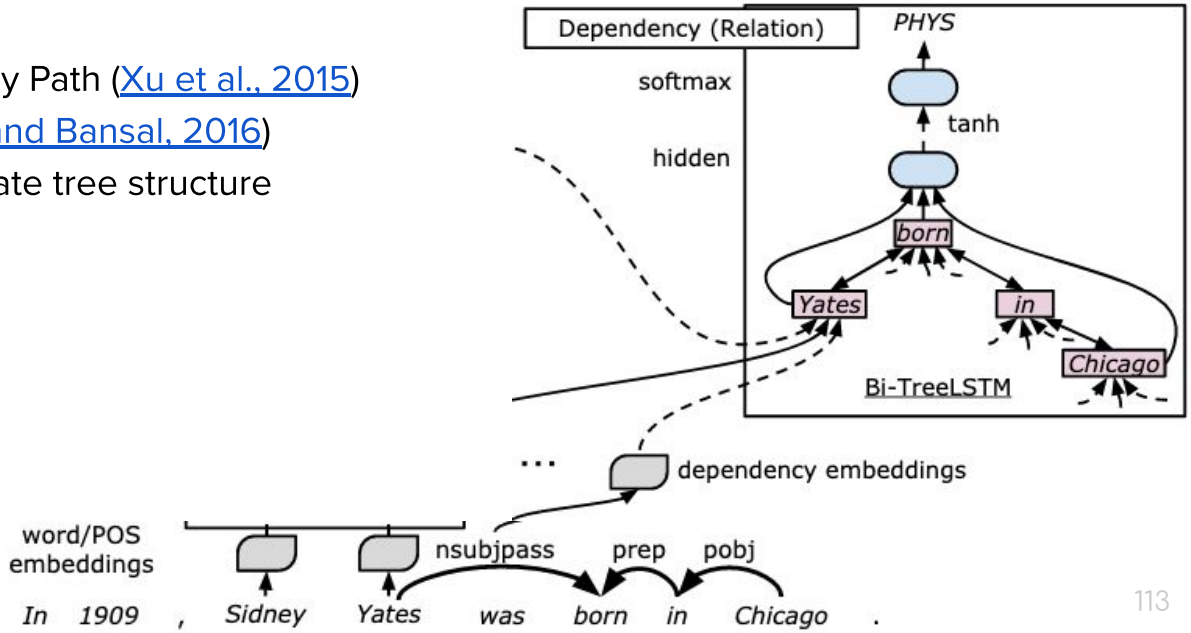


Relation Extraction: Neural Approaches

- Using **position features**
  - CNN + positions (Zeng et al., 2014)
  - RNN + max pooling + **entity markers** (Zhang and Wang, 2015)

- Using **tree-structures**
  - LSTM + Shortest Dependency Path (Xu et al., 2015)
  - BiLSTM + Tree LSTM (Miwa and Bansal, 2016)
    - ➜ Selecting the appropriate tree structure

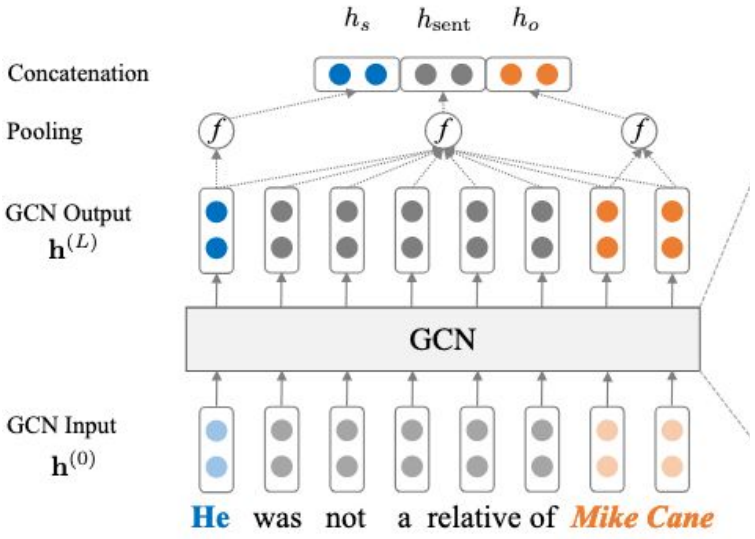




Relation Extraction: Neural Approaches

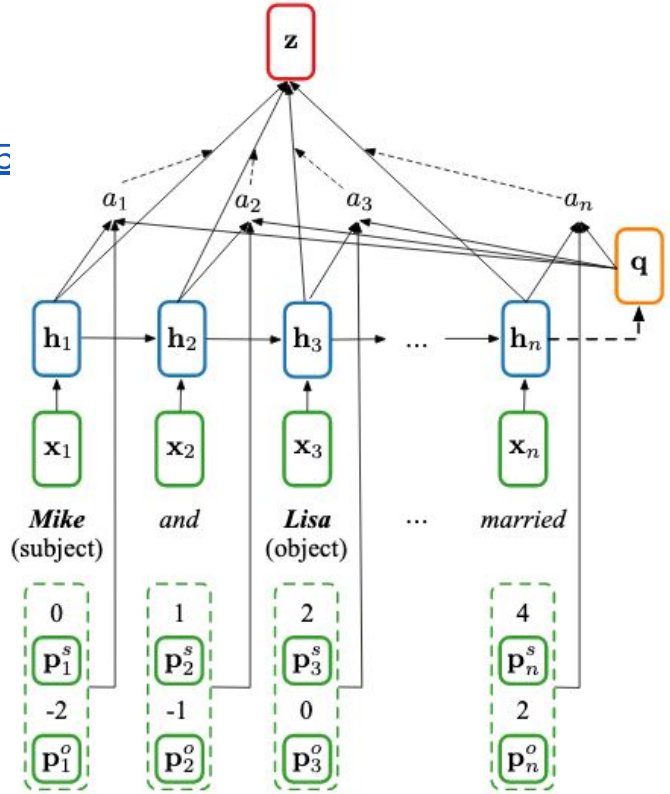
- Using **position features**
  - CNN + positions (Zeng et al., 2014)
  - RNN + max pooling + **entity markers** (Zhang and Wang, 2015)

- Using **tree-structures**
  - LSTM + Shortest Dependency Path (Xu et al., 2015)
  - BiLSTM + Tree LSTM (Miwa and Bansal, 2016)
    - ➜ Selecting the appropriate tree structure
  - GCN on trees (Zhang et al., 2018)
    - ➜ Not all edges are important
  - Graph LSTM (Song et al., 2018)



Relation Extraction: Neural Approaches

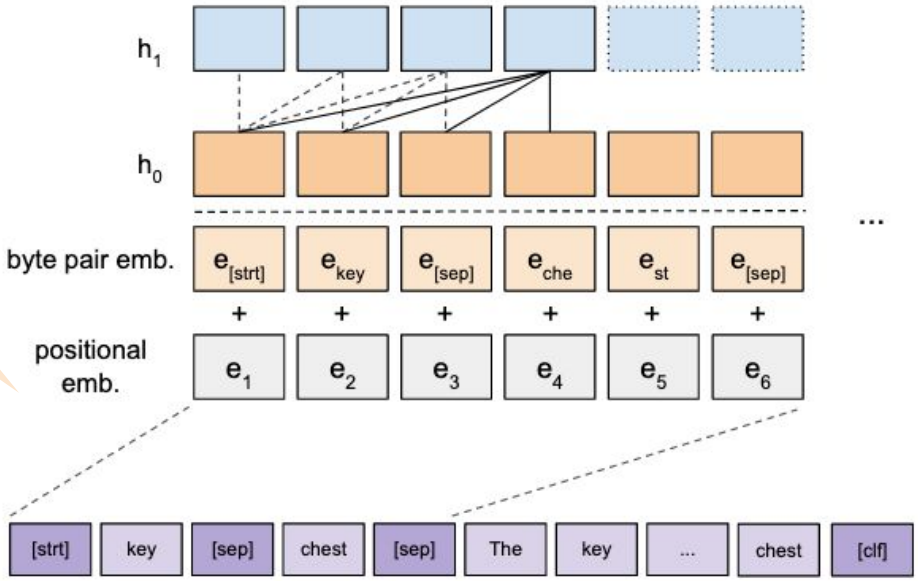
- Using **position features**
  - CNN + positions (Zeng et al., 2014)
  - RNN + max pooling + **entity markers** (Zhang and Wang, 2015
- Using **tree-structures**
  - LSTM + Shortest Dependency Path (Xu et al., 2015)
  - BiLSTM + Tree LSTM (Miwa and Bansal, 2016)
    - ➜ Selecting the appropriate tree structure is important
  - GCN on trees (Zhang et al., 2018)
    - ➜ Not all edges are important
  - Graph LSTM (Song et al., 2018)
- Using **attention**
  - CNN + Attention (Wang et al., 2016)
  - BiLSTM + Attention (Zhou et al., 2016)
  - BiLSTM + Attention + Relative Positions (Zhang et al., 2017)



Relation Extraction: Pre-trained LMs

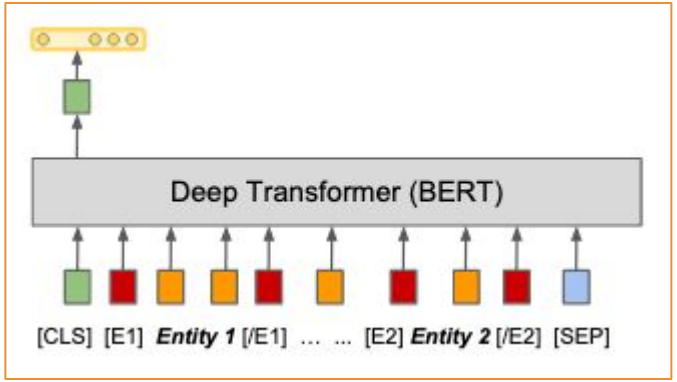
- GPT-2 (Alt et al., 2019)

Arguments are appended at the beginning to attend to the rest of the sentence



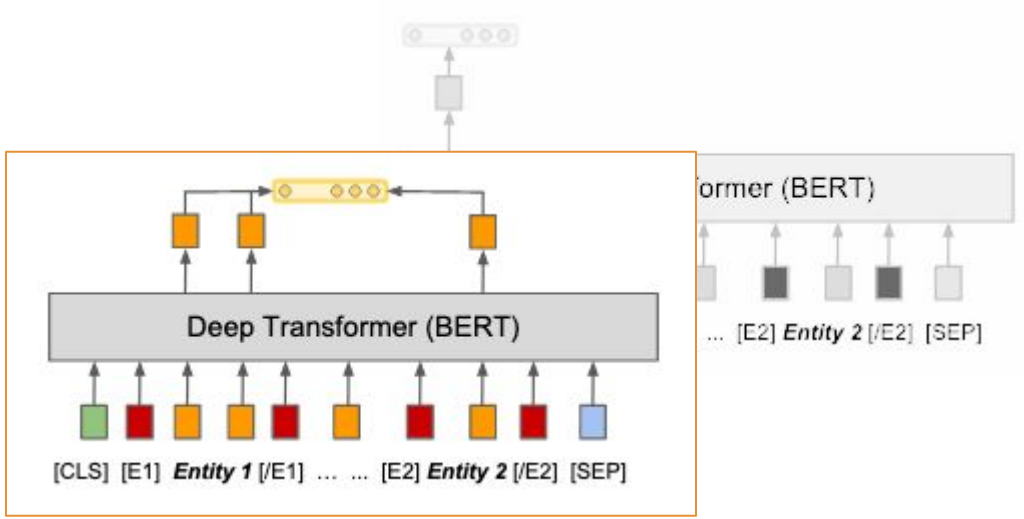
Relation Extraction: Pre-trained LMs

- GPT-2 (Alt et al., 2019)
- BERT (Soares et al., 2019)
  - Task as “fill-in the blanks”



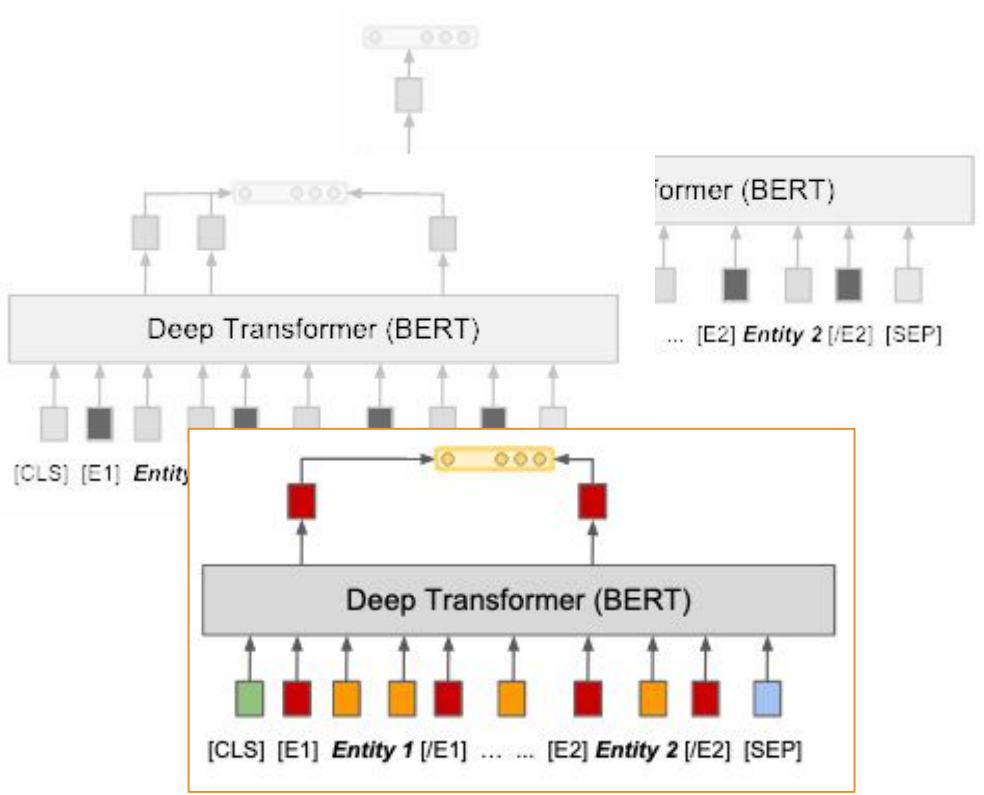
Relation Extraction: Pre-trained LMs

- GPT-2 (Alt et al., 2019)
- BERT (Soares et al., 2019)
  - Task as “fill-in the blanks”



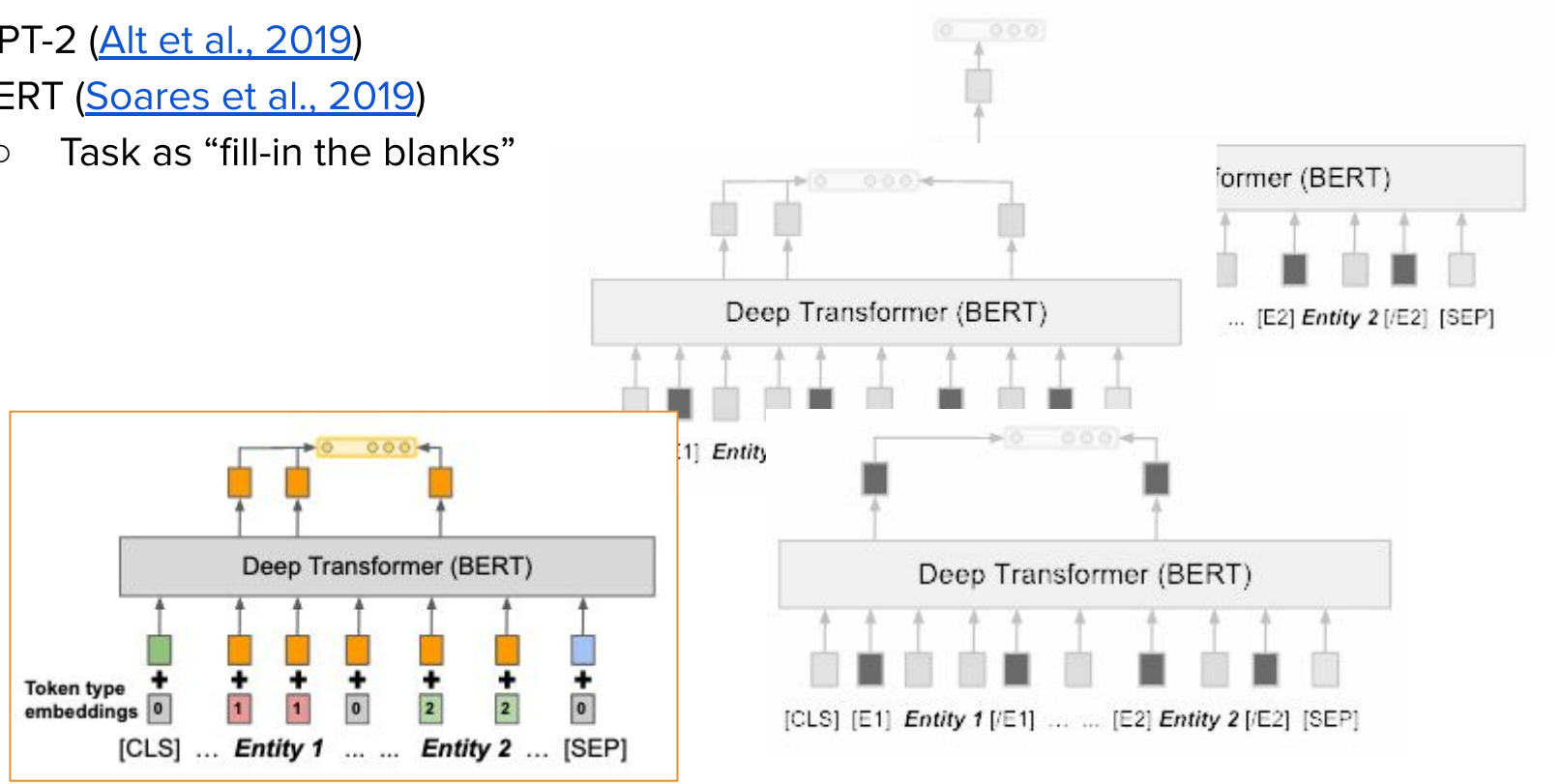
Relation Extraction: Pre-trained LMs

- GPT-2 (Alt et al., 2019)
- BERT (Soares et al., 2019)
  - Task as “fill-in the blanks”



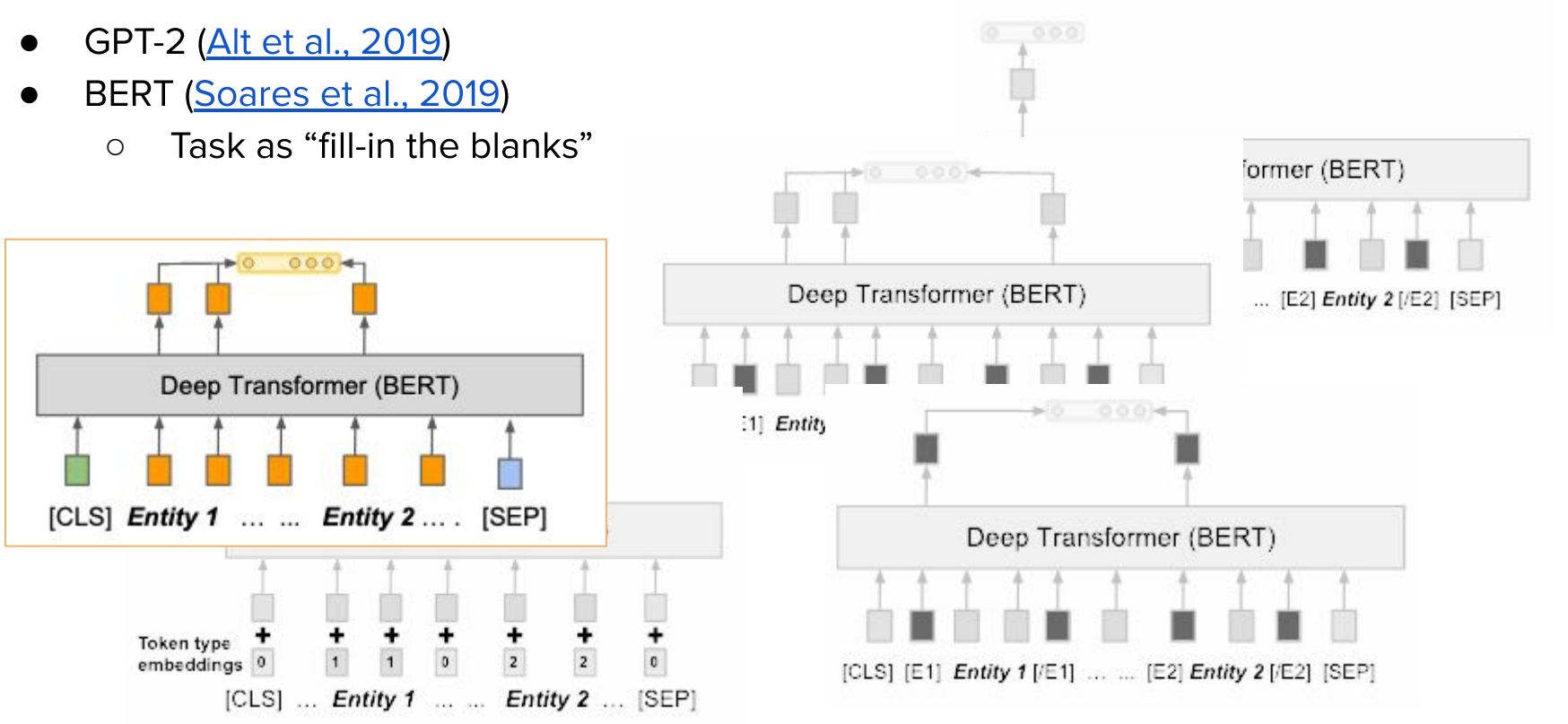
Relation Extraction: Pre-trained LMs

- GPT-2 (Alt et al., 2019)
- BERT (Soares et al., 2019)
  - Task as “fill-in the blanks”

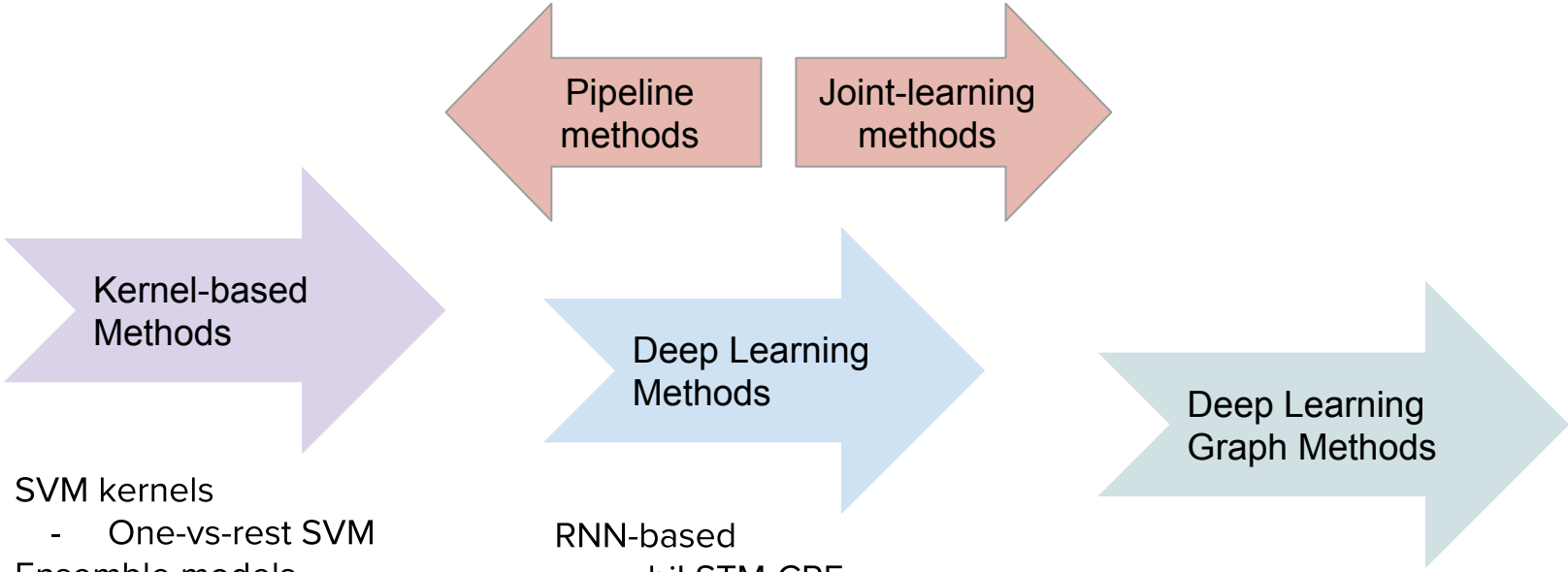


Relation Extraction: Pre-trained LMs

- GPT-2 (Alt et al., 2019)
- BERT (Soares et al., 2019)
  - Task as “fill-in the blanks”



Event Extraction



Event Extraction

General assumption: EE can be broken down to sub-tasks:
NER | Trigger detection | Argument detection | Flat event prediction | Nested event prediction | Metaknowledge

Pipeline methods
train a set of independent classifiers for each subtask

- Modular approach
- Assumption of subtask independence
- Error propagation

Joint-learning methods
learn (some) subtasks jointly

- Shared parameters
- Assumption of subtask correlation
- Transformation of initial subtasks
  - Link prediction - MLN (Vlachos et al, 2011)
  - Information networks (Li et al., 2014)
  - Joint learning trigger-relation-EE with shared CNN parameters

Event Extraction

Kernel-based methods:

SVMs with different feature-sets:

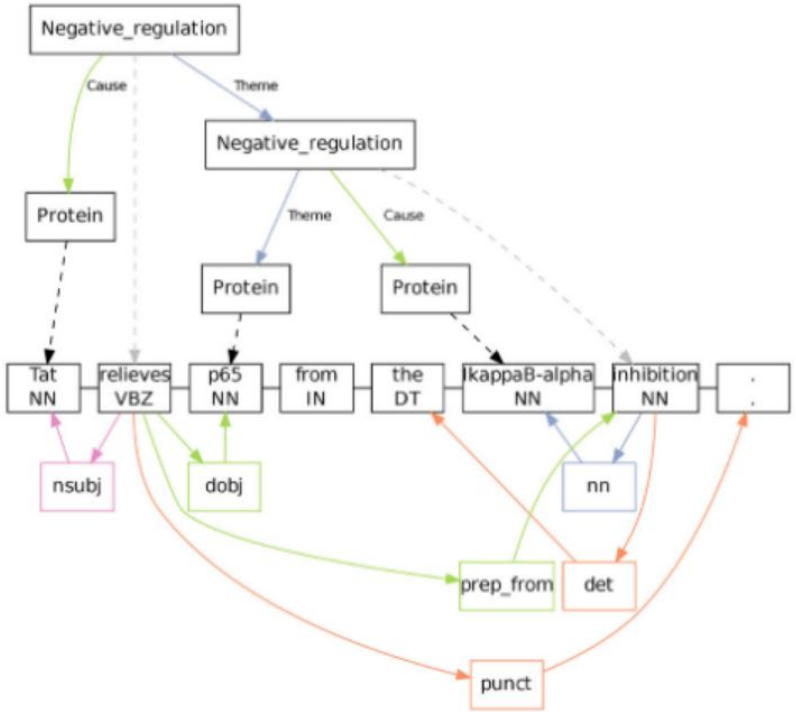
(Bjorne et. al, 2013, Miwa et al., 2013)

- Dependency parsing - tree structures
- Syntactic parsing
- Semantic features

\+ External resources (Miwa et. al, 2012)

Ensemble methods:

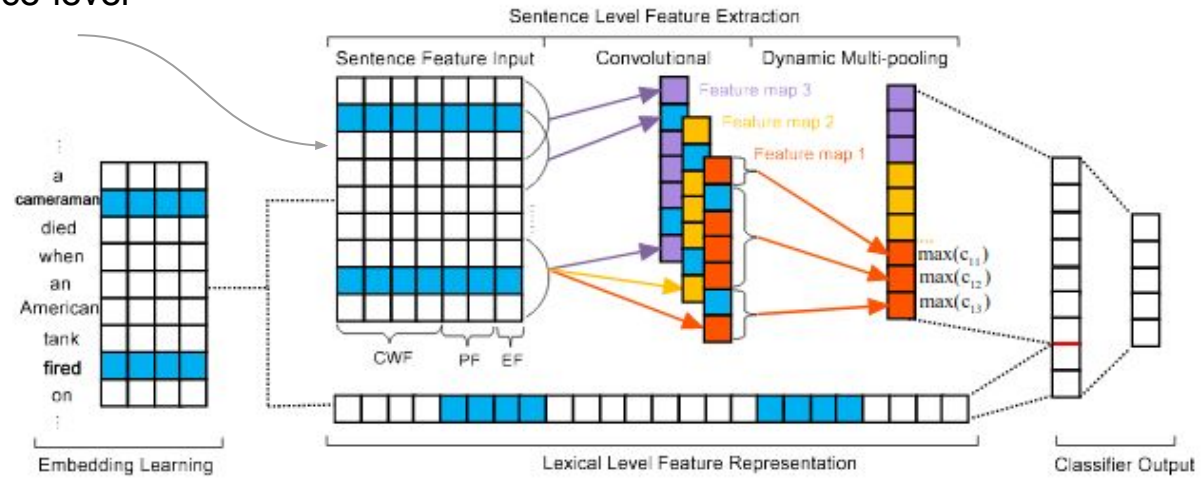
(Bali et al., 2020)



Event Extraction: Neural methods

- Pipelined architectures:
  - CNN with dynamic pooling (Chen et al., 2015)

Capture sentence-level features



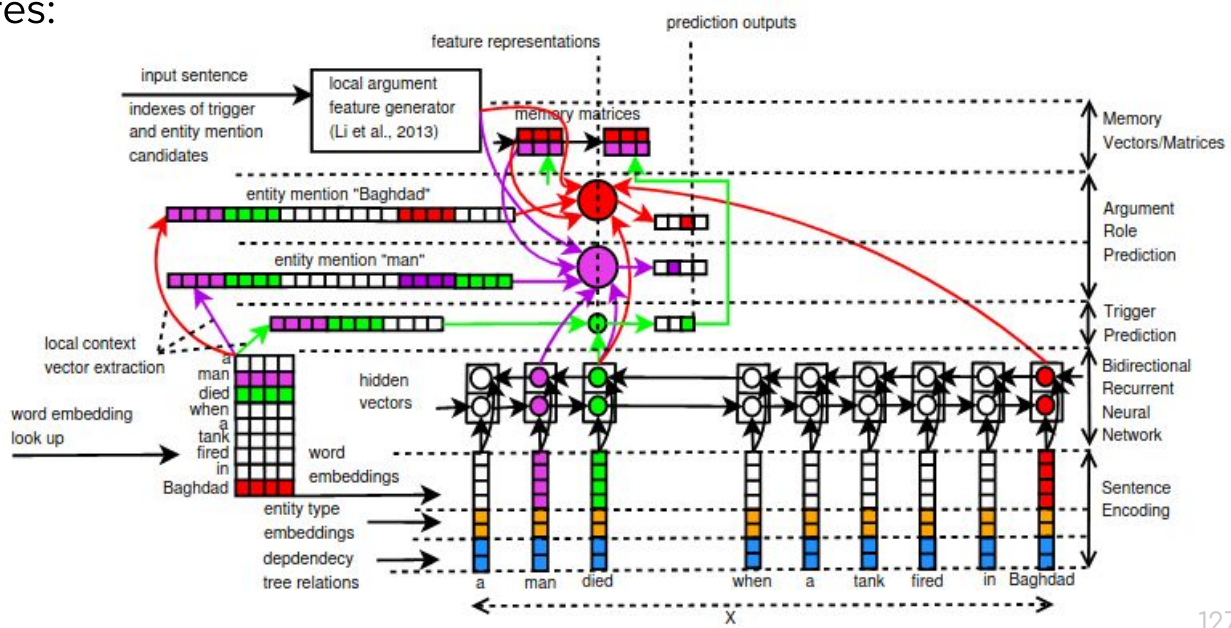
Event Extraction: Neural methods

- Pipelined architectures:
  - CNN with dynamic pooling (Chen et al., 2015)
  - CNN with dependency path embeddings (Bjorne et al., 2018)

Event Extraction: Neural methods

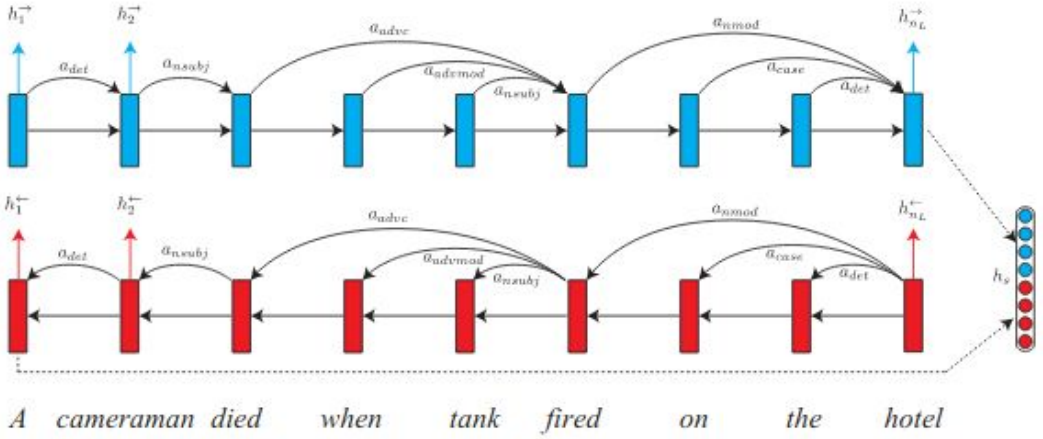
- Pipelined architectures:
  - CNN with dynamic pooling (Chen et al., 2015):
  - CNN with dependency path embeddings (Bjorne et al., 2018)
- Joint-learning architectures:
  - RNN with memory (Nguyen et al., 2016)

Assumption of task dependencies
- among trigger types
- among role types
- between argument roles and trigger subtypes



Event Extraction: Neural methods

- Pipelined architectures:
  - CNN with dynamic pooling (Chen et al., 2015):
  - CNN with dependency path embeddings (Bjorne et al., 2018)
- Joint-learning architectures:
  - RNN with memory (Nguyen et al., 2016)
  - RNN with dependency “Bridges” (Sha et al., 2018)



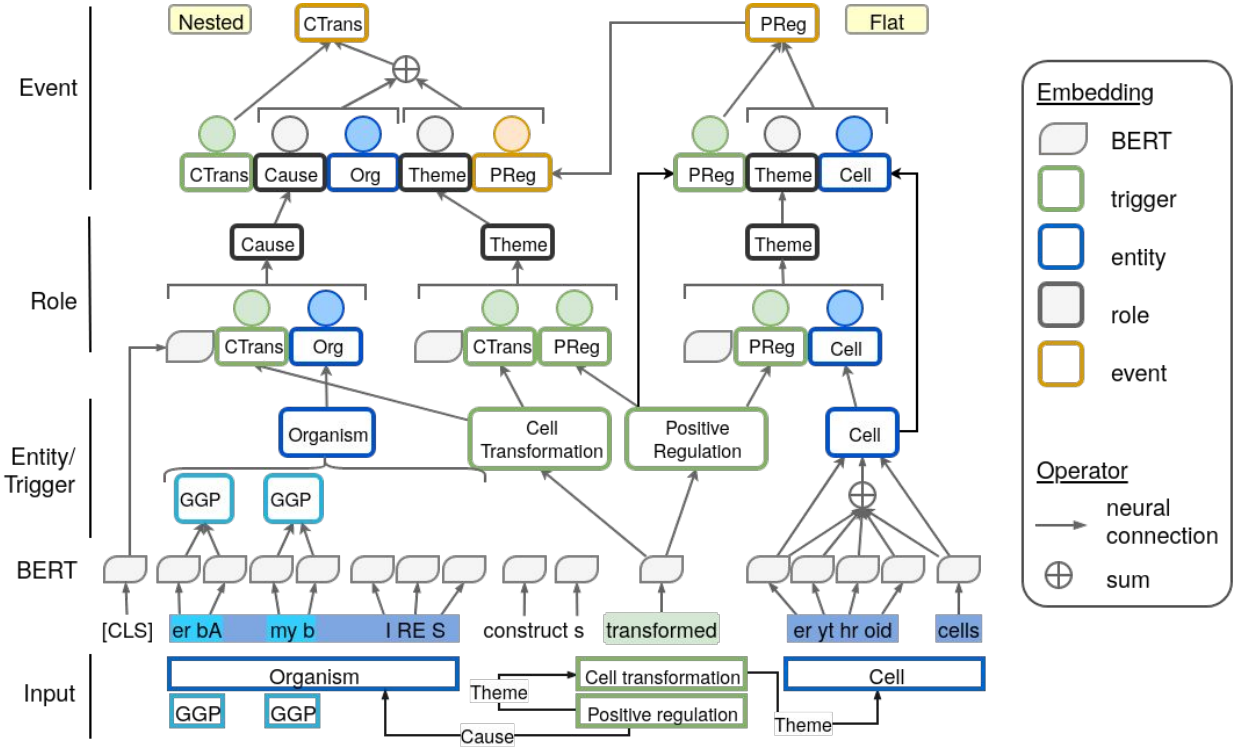
Event Extraction: Neural methods

- Pipelined architectures:
  - CNN with dynamic pooling (Chen et al., 2015):
  - CNN with dependency path embeddings (Bjorne et al., 2018)
- Joint-learning architectures:
  - RNN with memory (Nguyen et al., 2016)
  - RNN with dependency "Bridges" (Sha et al., 2018)
  - CNN with dependency embeddings (Li et. al, 2020)

Event Extraction:

BERT-based (Trieu et. al, 2020)

- Two settings:
  - Pipelined
  - Shared parameters
- Sub-word embeddings



Hands-on !


try:
start = date(int(self.start_year.
self.months.index(se
int(self.start_day.g
d = date(int(self.end_year.get(
self.months.index(se
int(self.end_day.ge
</ CODE>

Subword Segmentation: Byte Pair Encoding

- **Learnable subword segmentation →** Byte Pair Encoding (BPE)
  - Introduced by Sennrich et al. (2016)
  - It originates in data compression (Gage, 1994)
  - Replace common pairs of consecutive bytes with a byte that does not appear

This is a method to learn how to tokenize
NOT learn embeddings

- In order to perform subword tokenization BPE is modified
- Goal is to represent **rare** and **unseen words** as a sequence of **subword units**
- Hybrid between character- and word-level representations

Frequently occurring *subword pairs* are *merged* together

athazagoraphobia
ath | az | agor | aphobia

Byte Pair Encoding: Algorithm

1. Initialize the vocabulary, e.g. 100,000 words
2. Split each word into **characters**
3. Iterate:
   a. Count the **frequency** of each **consecutive pair of characters**
   b. **Merge** the **most frequent** tokens into a single one

pairs	*count*
a, t	19
m, a	10
t, s	2
t, e	6
e, a	2

word	*count*
c a t	4
m a t	5
m a t s	2
m a t e	3
a t e	3
e a t	2

Merges (total 5):

a t → at

* Credits to: https://lena-voita.github.io/nlp_course/seq2seq_and_attention.html#research_bpe_dropout

Byte Pair Encoding: Algorithm

1. Initialize the vocabulary, e.g. 100,000 words
2. Split each word into **characters**
3. Iterate:
   a. Count the **frequency** of each **consecutive pair of characters**
   b. **Merge** the **most frequent** tokens into a single one

pairs	*count*
a, t	19
m, a	10
t, s	2
t, e	6
e, a	2

word	*count*
c at	4
m at	5
m at s	2
m at e	3
at e	3
e at	2

Merges (total 5):

a t → at

* Credits to: https://lena-voita.github.io/nlp_course/seq2seq_and_attention.html#research_bpe_dropout

Byte Pair Encoding: Algorithm

1. Initialize the vocabulary, e.g. 100,000 words
2. Split each word into **characters**
3. Iterate:
   a. Count the **frequency** of each **consecutive pair of characters**
   b. **Merge** the **most frequent** tokens into a single one

pairs	*count*
c, at	4
m, at	10
at, s	2
at, e	6
e, at	2

word	*count*
c at	4
m at	5
m at s	2
m at e	3
at e	3
e at	2

Merges (total 5):

a t → at
m at → mat

* Credits to: https://lena-voita.github.io/nlp_course/seq2seq_and_attention.html#research_bpe_dropout

Byte Pair Encoding: Algorithm

1. Initialize the vocabulary, e.g. 100,000 words
2. Split each word into **characters**
3. Iterate:
   a. Count the **frequency** of each **consecutive pair of characters**
   b. **Merge** the **most frequent** tokens into a single one

pairs	*count*
c, at	4
m, at	10
at, s	2
at, e	6
e, at	2

word	*count*
c at	4
mat	5
mat s	2
mat e	3
at e	3
e at	2

Merges (total 5):

a t → at
m at → mat

* Credits to: https://lena-voita.github.io/nlp_course/seq2seq_and_attention.html#research_bpe_dropout

Byte Pair Encoding: Algorithm

1. Initialize the vocabulary, e.g. 100,000 words
2. Split each word into **characters**
3. Iterate:
   a. Count the **frequency** of each **consecutive pair of characters**
   b. **Merge** the **most frequent** tokens into a single one

pairs	*count*
c, at	4
mat, s	2
mat, e	3
at, e	3
e, at	2

word	*count*
c at	4
mat	5
mat s	2
mat e	3
at e	3
e at	2

Merges (total 5):

a t → at
m at → mat
c at → cat

* Credits to: https://lena-voita.github.io/nlp_course/seq2seq_and_attention.html#research_bpe_dropout

Byte Pair Encoding: Algorithm

1. Initialize the vocabulary, e.g. 100,000 words
2. Split each word into **characters**
3. Iterate:
   a. Count the **frequency** of each **consecutive pair of characters**
   b. **Merge** the **most frequent** tokens into a single one

pairs	*count*
c, at	4
mat, s	2
mat, e	3
at, e	3
e, at	2

word	*count*
cat	4
mat	5
mat s	2
mat e	3
at e	3
e at	2

Merges (total 5):

a t → at
m at → mat
c at → cat

* Credits to: https://lena-voita.github.io/nlp_course/seq2seq_and_attention.html#research_bpe_dropout

Byte Pair Encoding: Algorithm

1. Initialize the vocabulary, e.g. 100,000 words
2. Split each word into **characters**
3. Iterate:
   a. Count the **frequency** of each **consecutive pair of characters**
   b. **Merge** the **most frequent** tokens into a single one

pairs	*count*
mat, s	2
mat, e	3
at, e	3
e, at	2

word	*count*
cat	4
mat	5
mat s	2
mat e	3
at e	3
e at	2

Merges (total 5):

a t → at
m at → mat
c at → cat
mat e → mate

* Credits to: https://lena-voita.github.io/nlp_course/seq2seq_and_attention.html#research_bpe_dropout

Byte Pair Encoding: Algorithm

1. Initialize the vocabulary, e.g. 100,000 words
2. Split each word into **characters**
3. Iterate:
   a. Count the **frequency** of each **consecutive pair of characters**
   b. **Merge** the **most frequent** tokens into a single one

pairs	*count*
mat, s	2
mat, e	3
at, e	3
e, at	2

word	*count*
cat	4
mat	5
mat s	2
mate	3
at e	3
e at	2

Merges (total 5):

a t → at
m at → mat
c at → cat
mat e → mate

* Credits to: https://lena-voita.github.io/nlp_course/seq2seq_and_attention.html#research_bpe_dropout

Byte Pair Encoding: Algorithm

1. Initialize the vocabulary, e.g. 100,000 words
2. Split each word into **characters**
3. Iterate:
   a. Count the **frequency** of each **consecutive pair of characters**
   b. **Merge** the **most frequent** tokens into a single one

pairs	*count*
mat, s	2
at, e	3
e, at	2

word	*count*
cat	4
mat	5
mat s	2
mate	3
at e	3
e at	2

Merges (total 5):

a t → at
m at → mat
c at → cat
mat e → mate
at e → ate

* Credits to: https://lena-voita.github.io/nlp_course/seq2seq_and_attention.html#research_bpe_dropout

Byte Pair Encoding: Algorithm

1. Initialize the vocabulary, e.g. 100,000 words
2. Split each word into **characters**
3. Iterate:
   a. Count the **frequency** of each **consecutive pair of characters**
   b. **Merge** the **most frequent** tokens into a single one

pairs	*count*
mat, s	2
at, e	3
e, at	2

word	*count*
cat	4
mat	5
mat s	2
mate	3
ate	3
e at	2

Merges (total 5):

a t → at
m at → mat
c at → cat
mat e → mate
at e → ate

* Credits to: https://lena-voita.github.io/nlp_course/seq2seq_and_attention.html#research_bpe_dropout

Byte Pair Encoding: Algorithm

1. Initialize the vocabulary, e.g. 100,000 words
2. Split each word into **characters**
3. Iterate:
   a. Count the **frequency** of each **consecutive pair of characters**
   b. **Merge** the **most frequent** tokens into a single one

We add the ## symbol to distinguish segmented words

LIMIT REACHED

Final vocabulary:

cat
mat
mat## s
mate
at
e## at

word	*count*
cat	4
mat	5
mat s	2
mate	3
ate	3
e at	2

Merges (total 5):

a t → at
m at → mat
c at → cat
mat e → mate
at e → ate

* Credits to: https://lena-voita.github.io/nlp_course/seq2seq_and_attention.html#research_bpe_dropout

Subword Segmentation: Unigram LM

- Drawbacks of **BPE**:
  - Ambiguous resulting vocabulary
  - What if there is more than one way to separate a word?

ma | t | m | a | t | m | at

Final vocabulary:

m, ma, at, a, t

- Solution: **Probabilistic Unigram Language Model** (Kudo, 2018)
  - Predict the probability of a certain output given an initial state (LM objective)
  - *Unigram LM*: only considers the probability of the next word

Choose the most *likely* subword instead of the most *frequent*

Bonus!
Proposed a **subword regularisation** to take into account multiple subword segmentations

  - Algorithm
    1. Define a vocabulary size and a seed set (e.g. the BPE output!)
    2. Find the probability of each subword (assume: subwords occur independently)
    3. Calculate the loss of a subword in case it was removed
    4. Keep the top X% subwords with the smallest loss (typical 80%)

repeat

Subword Segmentation: WordPiece

- Originally introduced by Schuster and Nakajima (2012) used in BERT LM (Devlin et al., 2018)
- In the middle between BPE and Unigram LM
  - **BPE**
    - ✓ Choose what to merge → based on frequency
    - ✓ Merging → based on frequency
  - **Unigram LM**
    - ✓ Choose what to merge → based on probability
    - ✓ Merging → based on probability
  - **WordPiece**
    - ✓ Choose what to merge → based on frequency
    - ✓ Merging → based on probability

Merge the pair of characters that will result in the largest likelihood increase

Thank you !

Fenia fenia.christopoulou@gmail.com
Chryssa chryssa.zrv@gmail.com