Imperial College London

Pangaea

Introduction to Natural Language Processing and its Application in Healthcare

Jingqing Zhang, PhD
Research Associate, Imperial College London
Head of AI, Pangaea Data Limited
Email: jingqing.zhang15@imperial.ac.uk

Imperial College London

Pangaea

Human Languages

- Language is the basis of most human communication and interaction. Also, language is central to human thought, to how social and emotional relations are formed, to how we identify ourselves socially and personally, and to how humans record knowledge and develop societal intelligence.

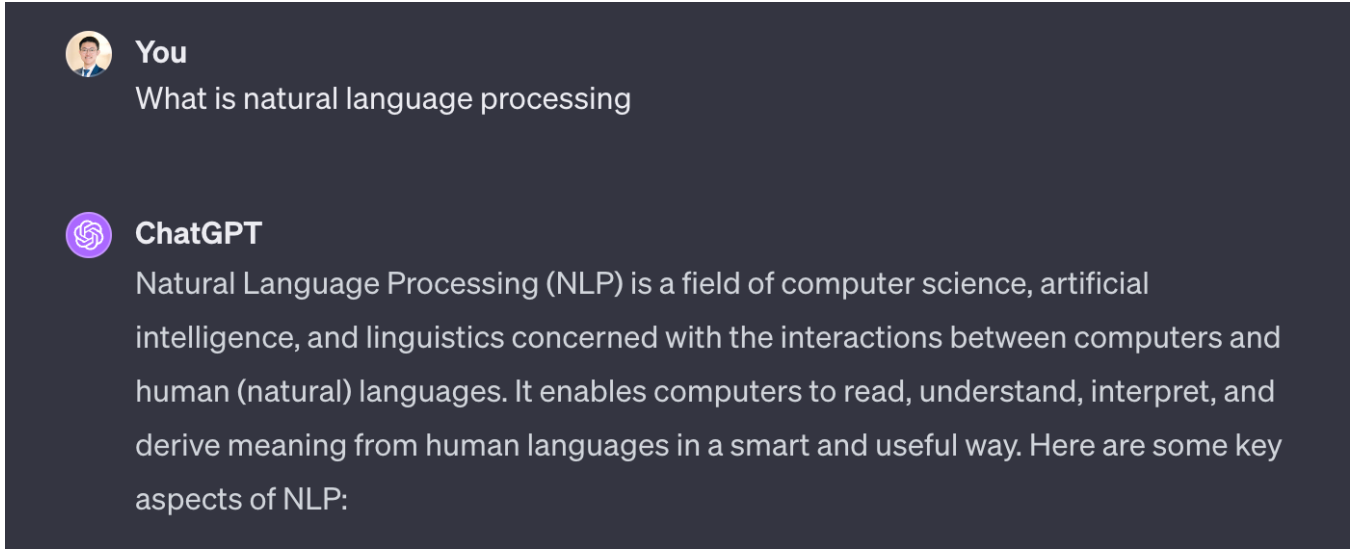
- Languages are complex yet efficient. It takes time for human to learn languages. Languages evolves through human history. Over 6,000 languages in the world. Languages are often ambiguous. Language data is large.

- Human ability of understanding and generating languages is general, flexible and robust, hence we hope computer can achieve a similar capability.

Imperial College London

Pangaea

Natural Language Processing (NLP)



- Definition by ChatGPT-4

Imperial College London

Pangaea

Natural Language Processing (NLP)

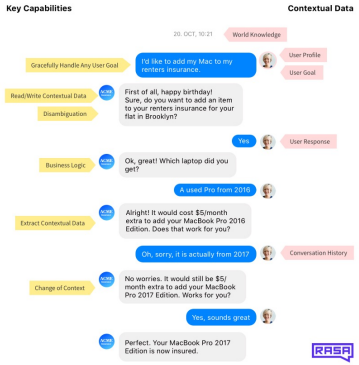
- Natural language processing (NLP) designs computational algorithms to process, parse, read, and generate human languages in much the same way human beings can.

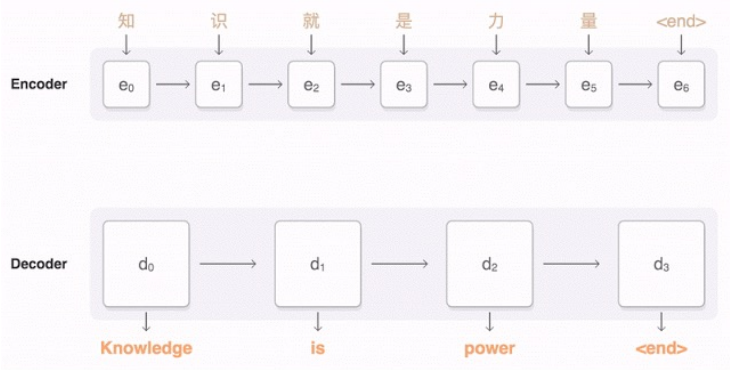
- NLP algorithms are usually designed for one specific linguistic task, but in recent years, more advanced NLP algorithms are designed for multi-tasking and combined with multiple-modal techniques (like vision and audio).

Imperial College London

Pangaea

What Tasks Can NLP Help?





- Understanding
- Sentiment analysis
- Topic Classification

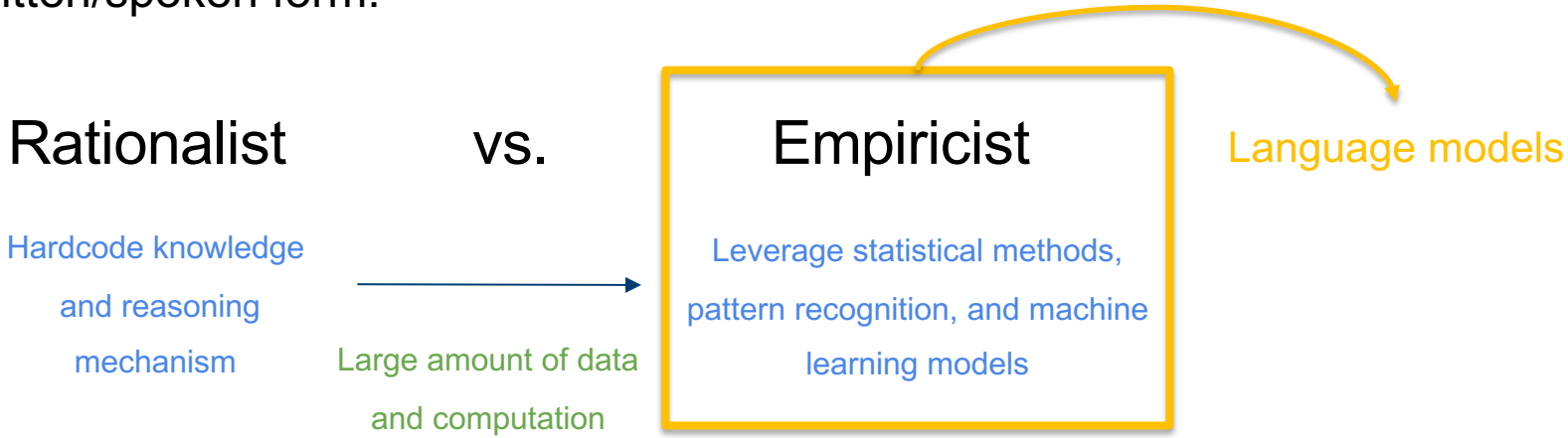
- Generation
- Chatbot
- Machine Translation

Imperial College London

Pangaea

How NLP Algorithms Have Been Designed?

- Many NLP algorithms are motivated by understanding the relationships between the mental representation of human languages and its manifestation in written/spoken form.



Imperial College London

Pangaea

Language Models

- In computer, text is a sequence of tokens: $X = \{w_1, w_2, \dots, w_N\}$
- Tokens can be words, sub-words, characters or even bytes.

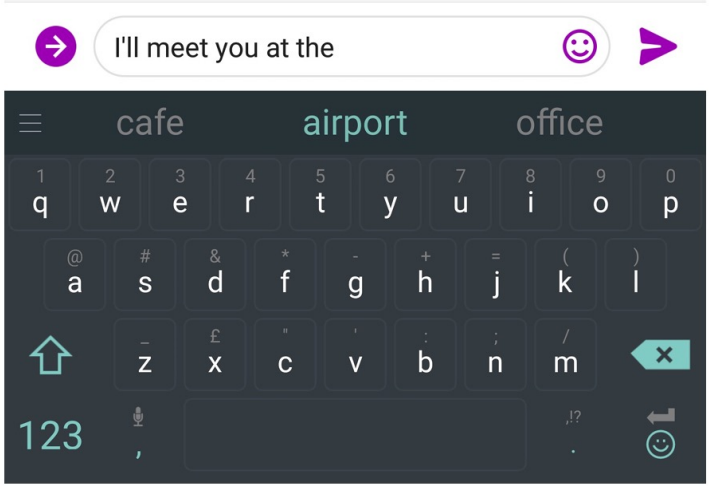
- A language model defines and learns a probability distribution over sequences of tokens in human languages, such as

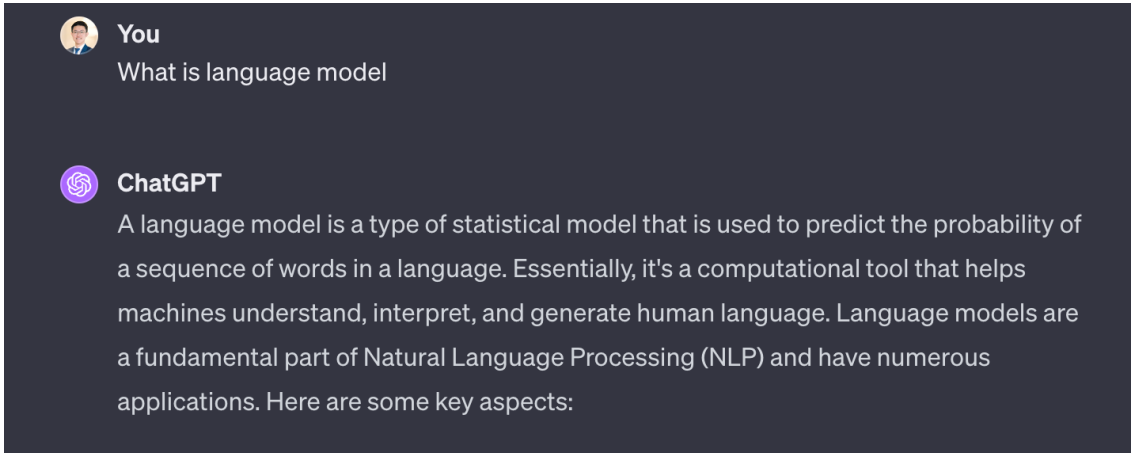
- $p(X) = p(w_1, w_2, \dots, w_n)$
- $p(X)$ should be high (close to 1) if X is human-readable and meaningful text
- $p(X)$ should be low (close to 0) if X is not readable nor meaningful

Imperial College London

Pangaea

Language Models are around us





- ChatGPT-4

Basic Language Models

Uni-gram $$p(w_1, w_2, \ldots, w_N) = \prod_{i=1}^{N} p(w_i)$$ Assume independency between w_i

Bi-gram $$p(w_1, w_2, \ldots, w_N) = \prod_{i=1}^{N} p(w_i|w_{i-1})$$ Assume w_i only depends on w_{i-1}

N-gram $$p(w_1, w_2, \ldots, w_N) = \prod_{i=1}^{N} p(w_i|w_{i-n+1}, \ldots, w_{i-1})$$

Imperial College London

Pangaea

N-gram Language Model

$$p(w_1, w_2, \ldots, w_N) = \prod_{i=1}^{N} p(w_i | w_{i-n+1}, \ldots, w_{i-1})$$

Training Data

Human language can be ambiguous.
Natural language process is complex.
Natural language process is difficult.
Natural language processing (NLP) designs computational algorithms to process, parse, read, and generate human languages in much the same way human beings can.

$p(ambiguous \mid human\ language\ can\ be) = 1$

$p(complex \mid natural\ language\ process\ is) = 0.5$

$p(challenging \mid natural\ language\ process\ is) = 0$

p is very likely to be zero especially when n is large and the tuple $(w_{i-n}, \ldots, w_i)$ never appears in the training data (sparsity problem), though "challenging" and "difficult" have similar meanings in this context.

Imperial College London

Pangaea

How to Represent Meanings of Words?

signifier (symbol) ⟺ signified (idea or thing)

= denotational semantics

tree ⟺ {  ,  ,  , …}

Imperial College London

Pangaea

How to Represent Meanings of Words in a Computer?

- Previously commonest NLP solution is to use, e.g., WordNet, a thesaurus containing lists of synonym sets and hypernyms (“is a” relationships).

- Synonyms: good, estimable, honorable, respectable, well
- Hypernyms: panda is a mammal, animal, living thing

- However, there is nuance. For example, “honorable” is listed as a synonym for “good” but this is only correct in some context.
- Meanings of words are updating. It requires human labour to create and adapt. It is subjective. It can not be used to accurately compute word similarity.

Imperial College London

Pangaea

Represent Word Meanings by Discrete Symbols

- In conventional NLP, we represent words as discrete symbols (localist representation)
- For example, we represent words by one-hot vectors
  - natural: [1,0,0,0,0,0,0,0,0,0]
  - language: [0,1,0,0,0,0,0,0,0,0]
  - processing: [0,0,1,0,0,0,0,0,0,0]

The dimension corresponds to the word is 1 and the rest is 0s.

- Vector dimension = number of words in vocabular $|V|$
- If the vocabulary size $|V|$ is very large like 100,000, the computational cost is very high (curse of dimensionality).
- Any two vectors are orthogonal. In other words, the product of two one-hot vectors always equals to 0. These can not be used to represent similarity between words.

Imperial College London

Pangaea

Represent Word Meanings by Context

- Distributional semantics: a word's meaning is given by the words that frequently appear close-by (i.e., the context of the word).

…government debt problems turning into	***banking***	*crises as happened in 2009…*
…saying that Europe needs unified	***banking***	*regulation to replace the hodgepodge…*
…India has just given its	***banking***	*system a shot in the arm…*

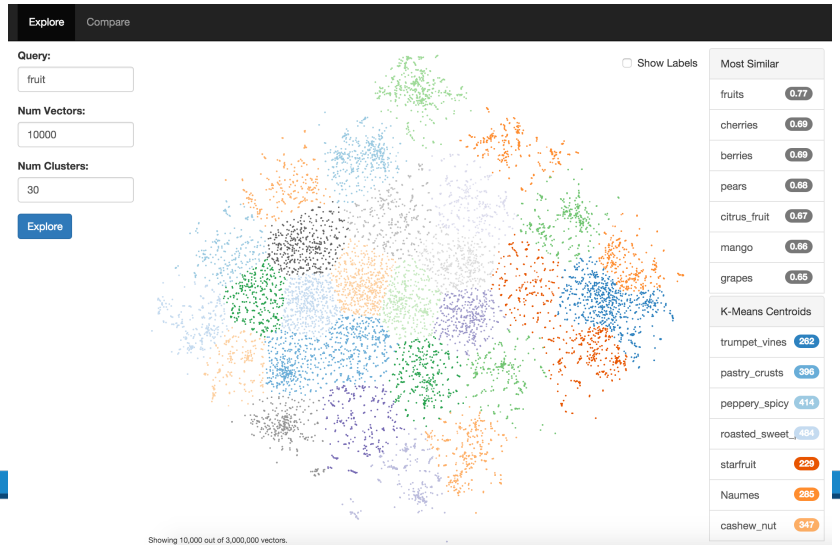
These context words will represent ***banking***

Imperial College London

Pangaea

Word Vectors

- We build a dense vector for each word.
- Similar word vectors if the words have similar context (statistically).
- Meaning similarity of words can be measured by the product of word vectors.

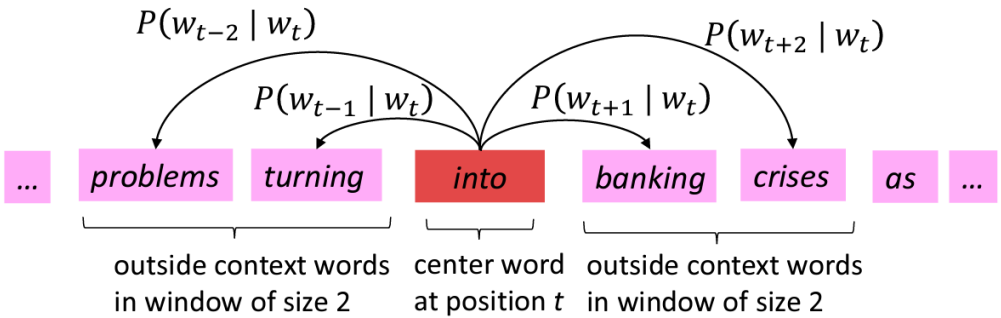


Imperial College London

Pangaea

How to Build Dense Word Vectors: Word2vec

- Given a large amount of text (corpus), every word is represented by a vector v_c
- Use the similarity of the word vectors for a central word c and its context o to calculate the probability $p(c|o)$ or $p(o|c)$
- Iteratively updating v_c to maximize $p(c|o)$ or $p(o|c)$



Imperial College London

Pangaea

Word2vec's Language Models

- Continuous Bag of Word Model (CBOW)

$$\max \quad P(c \mid o) = \quad P(w_c | w_{c-m}, \dots w_{c-1}, w_{c+1}, \dots, w_{c+m})$$

- Skip-gram Model:

$$\max \quad P(o \mid c) = \quad P(w_{c-m}, \dots w_{c-1}, w_{c+1}, \dots, w_{c+m} | w_c)$$

Imperial College London

Pangaea

Skip-gram Model

$$\max P(o \mid c) = \min J = -\log P(w_{c-m}, \dots w_{c-1}, w_{c+1}, \dots, w_{c+m} | w_c)$$

$$= -\log \prod_{j=0, j\neq m}^{2m} P(w_{c-m+j} \mid w_c)$$ Conditional independence

$$= -\log \prod_{j=0, j\neq m}^{2m} P(u_{c-m+j} \mid v_c)$$ Replace with vectors

$$= -\log \prod_{j=0, j\neq m}^{2m} \frac{\exp\left(u_{c-m+j}^T v_c\right)}{\sum_{k=1}^{|V|} \exp(u_k^T v_c)}$$ Softmax with dot-product

$$= -\sum_{j=0, j\neq m}^{2m} u_{c-m+j}^T v_c + 2m \log \sum_{k=1}^{|V|} \exp(u_k^T v_c)$$ Expand log

Negative Sampling

$$\min J = -\sum_{j=0, j\neq m}^{2m} u_{c-m+j}^T v_c + 2m \log \sum_{k=1}^{|V|} \exp(u_k^T v_c)$$

Expensive

D : set of positive samples

$\tilde{D}$: set of negative samples

With $|\tilde{D}|$ size much smaller than $|V|$

$$\min J = -\log \prod_{(o,c)\in D} P(D=1 \mid o,c) \prod_{(k,c)\in \tilde{D}} P(D=0 \mid w,c)$$

$$= -\sum_{(o,c)\in D} \log \sigma(u_o^T v_c) - \sum_{(k,c)\in \tilde{D}} \log \sigma(u_k^T v_c)$$

Imperial College London

Pangaea

Evaluating Word2Vec

- In NLP, evaluation typically includes intrinsic vs extrinsic evaluation.
- Intrinsic evaluation
  - Evaluation on the task itself, which can be intermediate
  - Fast to compute
- Extrinsic evaluation
  - Evaluation on a real task
  - Can take a long time to compute accuracy
  - If replacing exactly one subsystem with another improves accuracy → Win

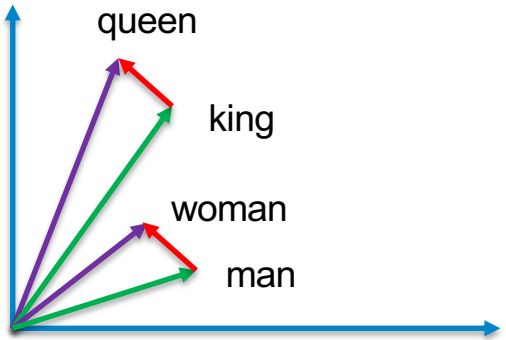
Imperial College London

Pangaea

Intrinsic Evaluation of Word2vec

- We evaluate how well the similarity of word vectors represents similarity of meanings

$$v(king) - v(man) + v(woman) = v(queen)$$



Imperial College London

Pangaea

Extrinsic Evaluation of Word2vec

- Named entity recognition (NER): identifying references to a person, location or organization.
- For example, “Imperial College London is in United Kingdom”

Model	Dev	Test	ACE	MUC7
Discrete	91.0	85.4	77.4	73.4
SVD	90.8	85.7	77.3	73.7
SVD-S	91.0	85.5	77.6	74.3
SVD-L	90.5	84.8	73.6	71.5
HPCA	92.6	**88.7**	81.7	80.7
HSMN	90.5	85.7	78.7	74.7
CW	92.2	87.4	81.7	80.2
CBOW	93.1	88.2	82.2	81.1
GloVe	**93.2**	88.3	**82.9**	**82.2**

Imperial College London

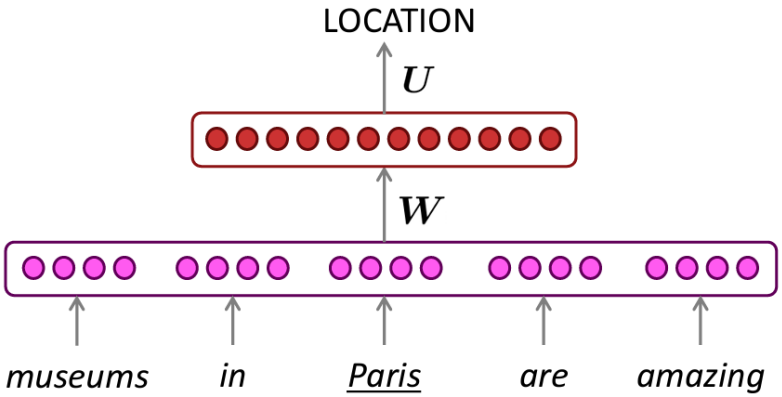
Pangaea

Use Word2vec in Neural Language Models for Named Entity Recognition (NER)

- Inputs: a sequence of words $w_1, w_2, \dots, w_t$
- Outputs: predict the entity type of a word w_i

- $p(\text{entity type of } w_i | \, w_1, w_2, \dots, w_t)$

Word vectors by word2vec (or other methods)



Imperial College London

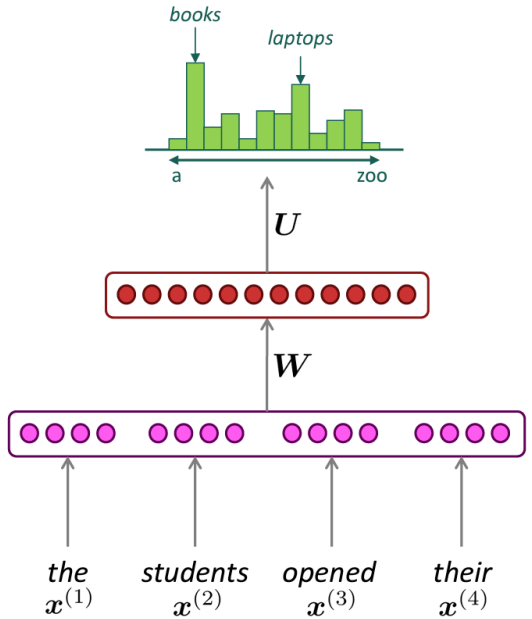
Pangaea

Use Word2vec in Neural Language Models to Generate Next Words

- Inputs: a sequence of words $w_1, w_2, \dots, w_t$
- Outputs: a sequence of the next word w_{t+1}

- $p(w_{t+1} \mid w_1, w_2, \dots, w_t)$

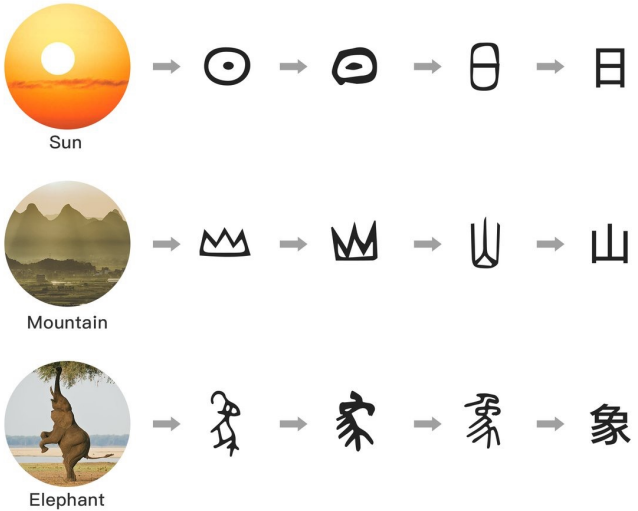
Word vectors by word2vec (or other methods)



Imperial College London

Pangaea

Other Ways to Represent Words



Imperial College London

Pangaea

Word2vec is not Perfect

- Each word is represented by one unique and fixed vector.
- But one word may have multiple meanings (polysemy).

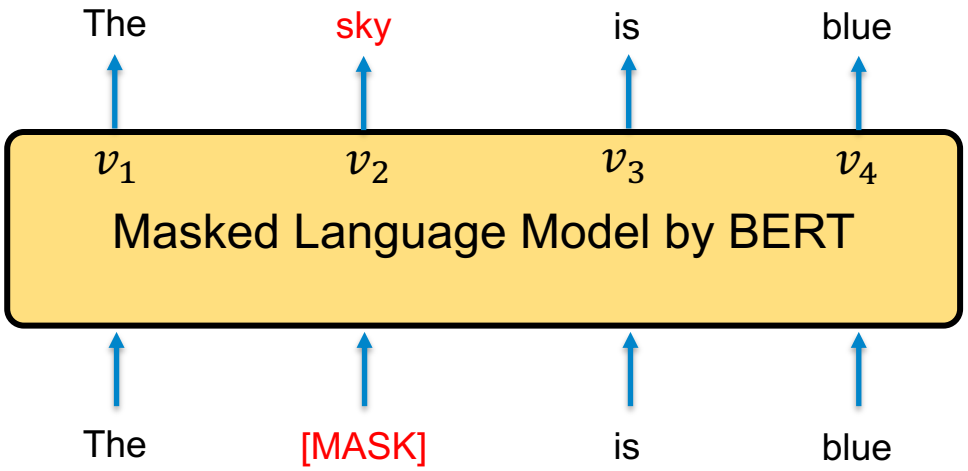
- Let's stick to the plan.
- I stick my fork into the sausage

- Context matters → *Contextualised* word vectors.
- Can we generate dynamic word vectors according to contexts in real time?

Imperial College London

Pangaea

Contextual Word Vectors by Masked Language Model



- We use context (both directions) to predict the target central word
- We obtain contextualized word vector v_i which encodes the central word as well as the present context

Imperial College London

Pangaea

Language Models can be Applied at Sentences

- Language models are not restricted to tokens/words.
- Some NLP tasks require language modelling for sentences. For example,

- Summarization is the process of condensing a large piece of text or information into a shorter, coherent, and concise version while retaining the key ideas and important details.

- Extractive Summarization: the summary is created by selecting and extracting sentences or phrases directly from the original text.
- Abstractive summarization involves generating new sentences that may not exist in the original text.

- Can we build sentence-level language models for abstractive summarization?

Imperial College London

Pangaea

Sentence-level Language Models for Summarization

- Gap Sentence Generation by PEGASUS which masks sentences and predicts sentences based on other unmasked contexts

Imperial College London

Pangaea

How to Evaluate Language Models?

- The standard intrinsic evaluation of language models is perplexity.

$$\text{perplexity} = \prod_{t=1}^{T} \left(\frac{1}{p(w_{t+1} \mid w_1, w_2, \ldots, w_t)} \right)^{1/T}$$

- Inverse probability of corpus
- Lower perplexity is better

Imperial College London

Pangaea

Larger Models tend to have Lower Perplexity

Hyperparams				Dev Set Accuracy		
#L	#H	#A	LM (ppl)	MNLI-m	MRPC	SST-2
3	768	12	5.84	77.9	79.8	88.4
6	768	3	5.24	80.6	82.2	90.7
6	768	12	4.68	81.9	84.8	91.3
12	768	12	3.99	84.4	86.7	92.9
12	1024	16	3.54	85.7	86.9	93.3
24	1024	16	3.23	86.6	87.8	93.7

Model size

Perplexity

Imperial College London

Pangaea

Larger Models tend to have Better Performance on Downstream Tasks on GLUE

System	MNLI-(m/mm)	QQP	QNLI	SST-2	CoLA	STS-B	MRPC	RTE	**Average**
	392k	363k	108k	67k	8.5k	5.7k	3.5k	2.5k	-
Pre-OpenAI SOTA	80.6/80.1	66.1	82.3	93.2	35.0	81.0	86.0	61.7	74.0
BiLSTM+ELMo+Attn	76.4/76.1	64.8	79.8	90.4	36.0	73.3	84.9	56.8	71.0
OpenAI GPT	82.1/81.4	70.3	87.4	91.3	45.4	80.0	82.3	56.0	75.1
$\text{BERT}_{\text{BASE}}$	84.6/83.4	71.2	90.5	93.5	52.1	85.8	88.9	66.4	79.6
$\text{BERT}_{\text{LARGE}}$	**86.7/85.9**	**72.1**	**92.7**	**94.9**	**60.5**	**86.5**	**89.3**	**70.1**	**82.1**

Imperial College London

Pangaea

Larger Models tend to have Better Performance on Downstream Tasks like Summarization

R1/R2/RL	Dataset size	$\text{Transformer}_{\text{BASE}}$	$\text{PEGASUS}_{\text{BASE}}$	Previous SOTA	$\text{PEGASUS}_{\text{LARGE}}$ (C4)	$\text{PEGASUS}_{\text{LARGE}}$ (HugeNews)
XSum	226k	30.83/10.83/24.41	39.79/16.58/31.70	45.14/22.27/37.25	45.20/22.06/36.99	**47.21/24.56/39.25**
CNN/DailyMail	311k	38.27/15.03/35.48	41.79/18.81/38.93	**44.16**/21.28/40.90	43.90/21.20/40.76	**44.17/21.47/41.11**
NEWSROOM	1212k	40.28/27.93/36.52	42.38/30.06/38.52	39.91/28.38/36.87	**45.07/33.39/41.28**	**45.15/33.51/41.33**
Multi-News	56k	34.36/5.42/15.75	42.24/13.27/21.44	43.47/14.89/17.41	46.74/17.95/24.26	**47.52/18.72/24.91**
Gigaword	3995k	35.70/16.75/32.83	36.91/17.66/34.08	**39.14/19.92/36.57**	38.75/**19.96**/36.14	**39.12/19.86**/36.24
WikiHow	168k	32.48/10.53/23.86	36.58/15.64/30.01	28.53/9.23/26.54	**43.06/19.71/34.80**	41.35/18.51/33.42
Reddit TIFU	42k	15.89/1.94/12.22	24.36/6.09/18.75	19.0/3.7/15.1	**26.54/8.94/21.64**	**26.63/9.01/21.60**
BIGPATENT	1341k	42.98/20.51/31.87	43.55/20.43/31.80	37.52/10.63/22.79	**53.63/33.16/42.25**	53.41/32.89/42.07
arXiv	215k	35.63/7.95/20.00	34.81/10.16/22.50	41.59/14.26/23.55	**44.70/17.27/25.80**	**44.67/17.18/25.73**
PubMed	133k	33.94/7.43/19.02	39.98/15.15/25.23	40.59/15.59/23.59	**45.49/19.90/27.69**	45.09/19.56/27.42
AESLC	18k	15.04/7.39/14.93	34.85/18.94/34.10	23.67/10.29/23.44	**37.69/21.85/36.84**	37.40/21.22/36.45
BillSum	24k	44.05/21.30/30.98	51.42/29.68/37.78	40.80/23.83/33.73	**57.20**/39.56/**45.80**	**57.31/40.19/45.82**

Imperial College London

Pangaea

Large Language Models (LLMs)

- These motivate large language models.

- Recent large language models (LLMs) typically use the Transformer model
- Transformer has an Encoder and a Decoder

- LLMs may have both Encoder and Decoder (T5)
- Or Encoder only (BERT family)
- Or Decoder only (GPT family)

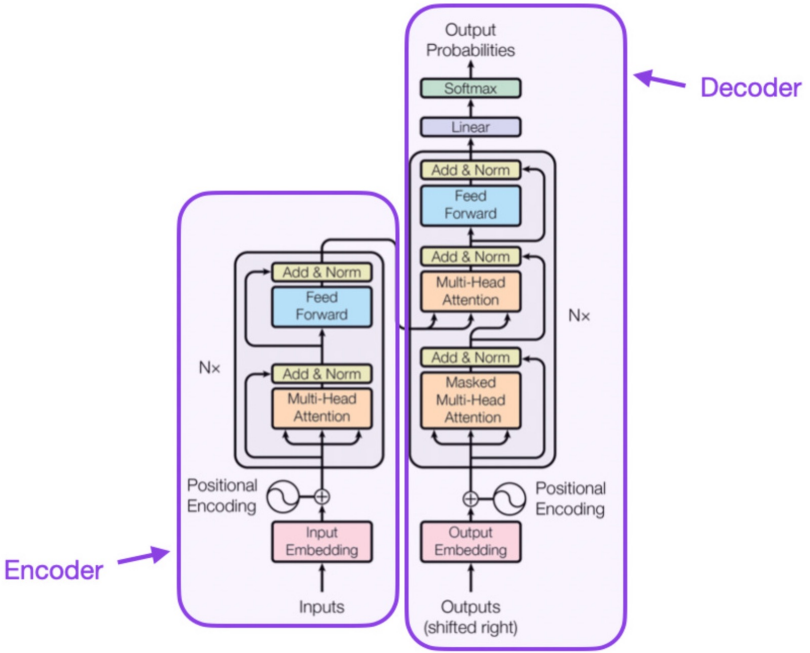
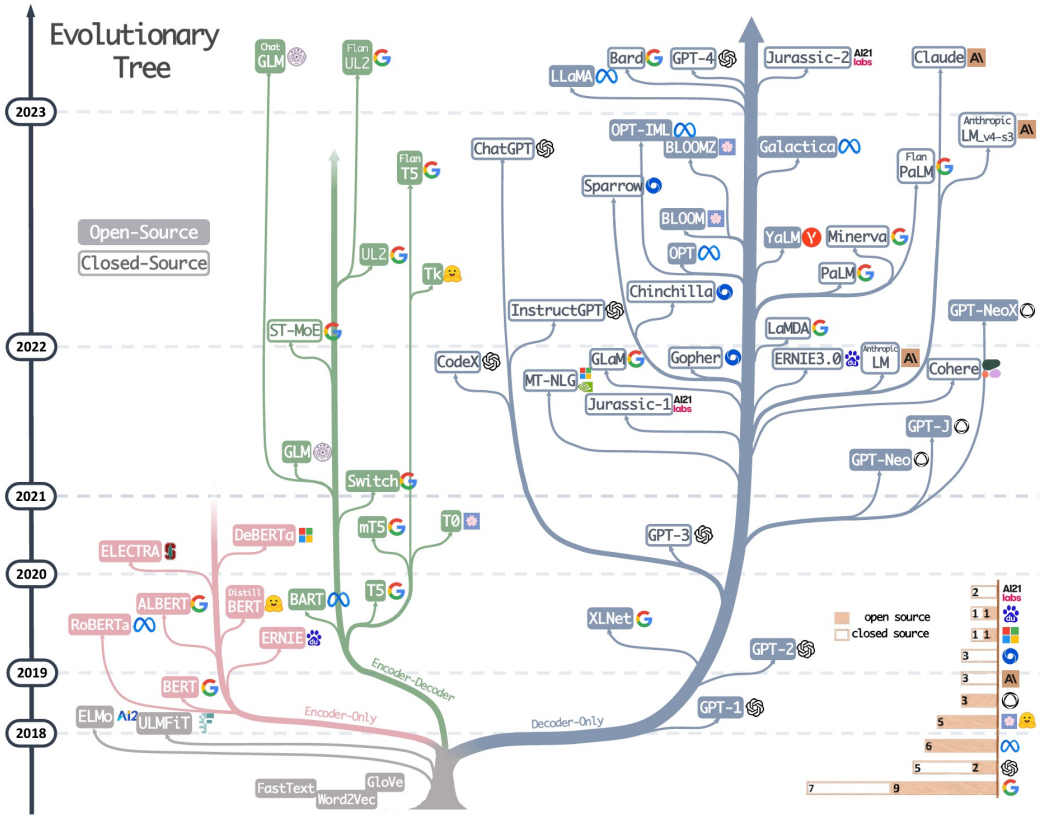


Figure 1: The Transformer - model architecture.

Imperial College London

Pangaea



Imperial College London

Pangaea

AI + NLP in Healthcare to Improve Patient Outcomes

- National Health Service (NHS) in UK captures billions of patient interactions each year.
- 80% of health data is unstructured textual data.

- Most cancer patients were first diagnosed at stage 3 or 4 and the survival rate for five years is very low.
- 50% of such patients can be diagnosed earlier in stage 1 or 2 if someone can read their primary care notes.

Imperial College London

Pangaea

Extract Clinical Features of Patients from Clinical Notes

“Patient A has hypertension and headache.”

“Her mother smokes but she denies smoking.”

“The patient has body temperature 40C.”

“The patient has history of back pain.”

Imperial College London

Pangaea

Extract Clinical Features from Patients from Clinical Notes

- Keyword based entity extraction is not enough.

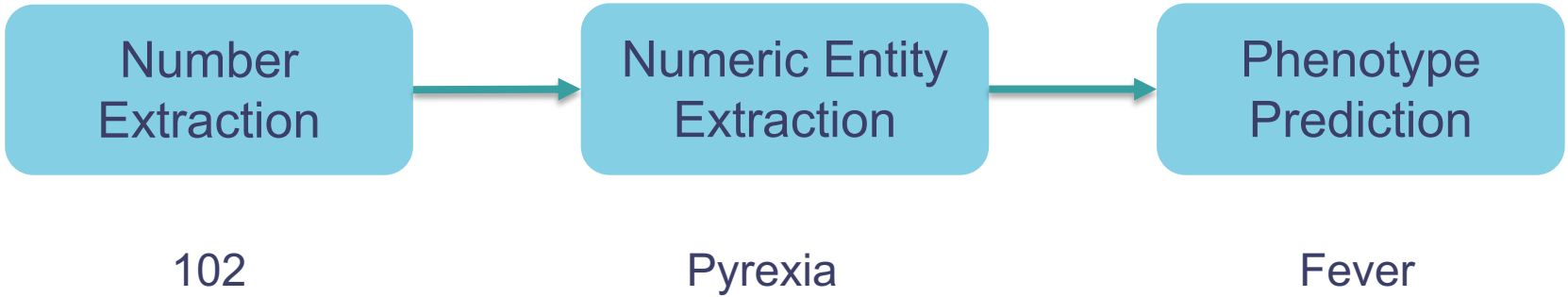
- Contextual semantics matters.
- Understand the numeric values.
- Disambiguate medical abbreviations. Pt → patient or physiotherapy

Imperial College London

Pangaea

Reasoning with Numbers in Clinical Notes

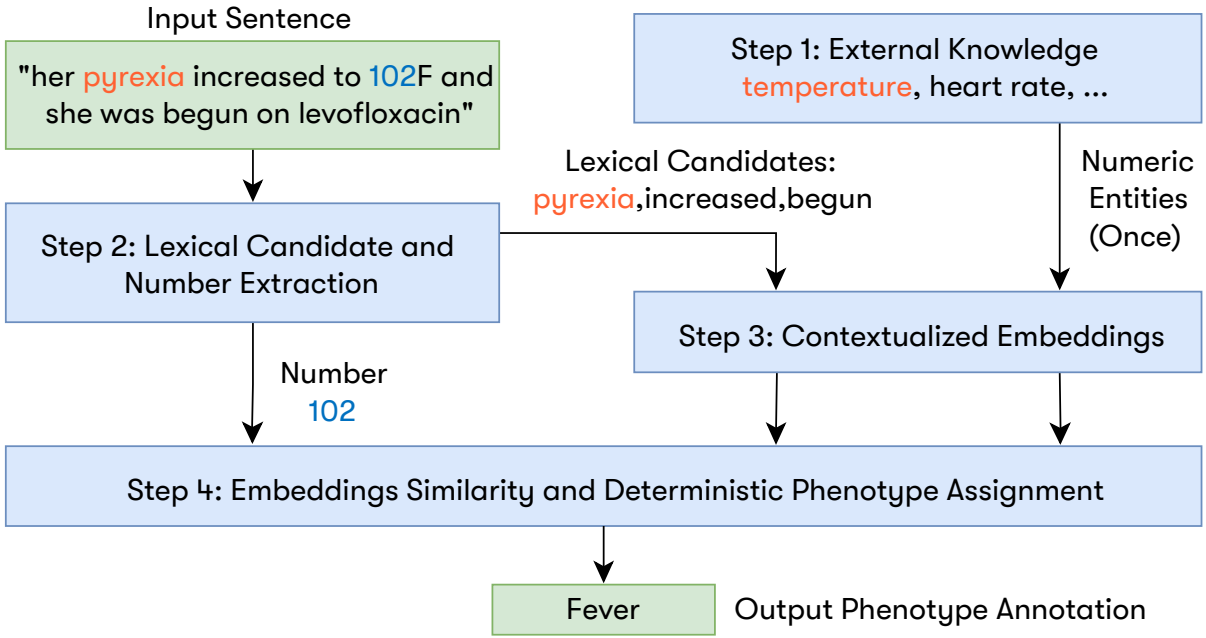
- “her pyrexia increased to 102F and she was begun on levofloxacin”
- We can formalize the numerical reasoning approach as:



Imperial College London

Pangaea

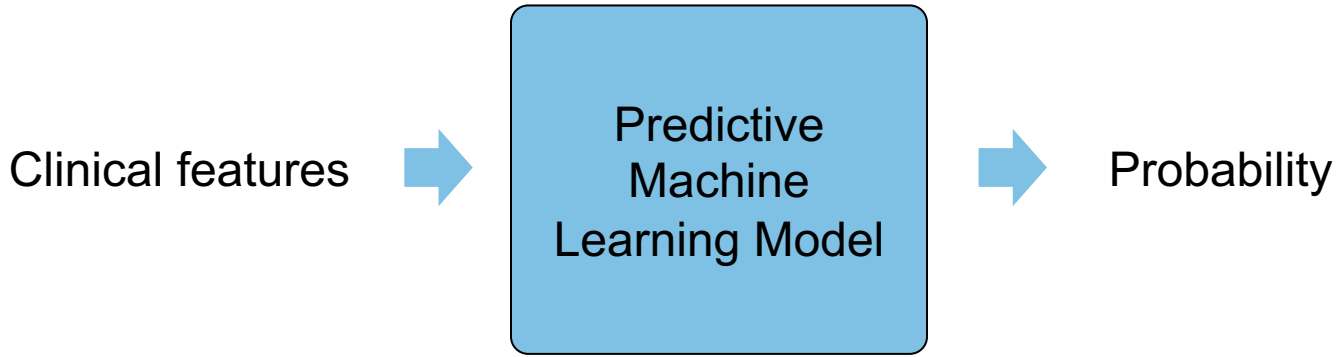
Reasoning with Numbers in Clinical Notes



Imperial College London

Pangaea

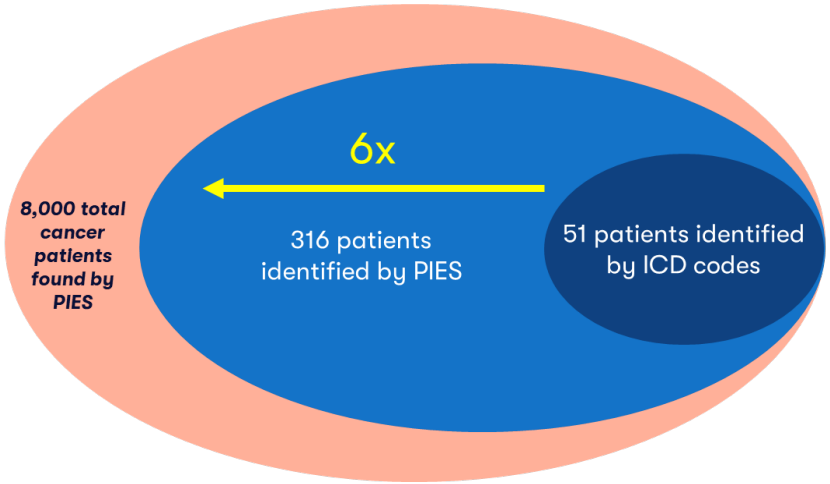
Find the Patient at Risk

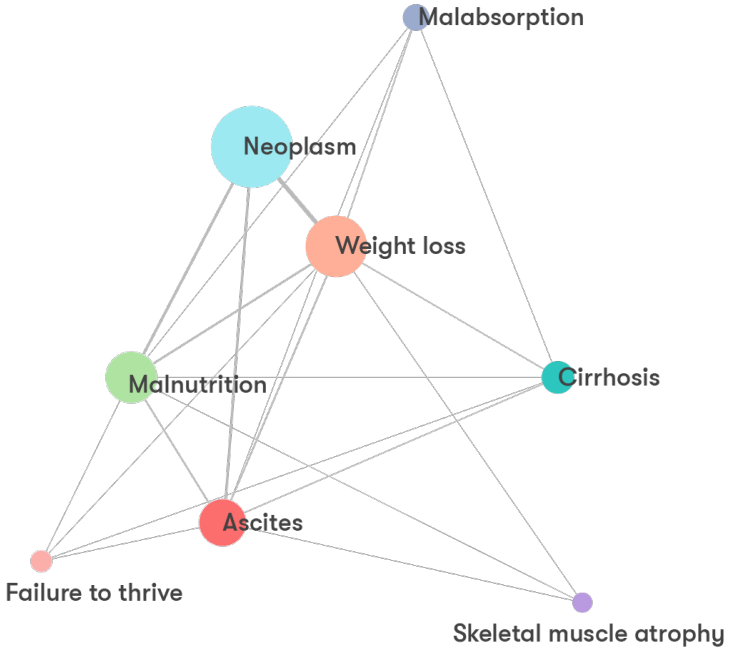


Imperial College London

Pangaea

Find 6x Patients Miscoded for Cachexia



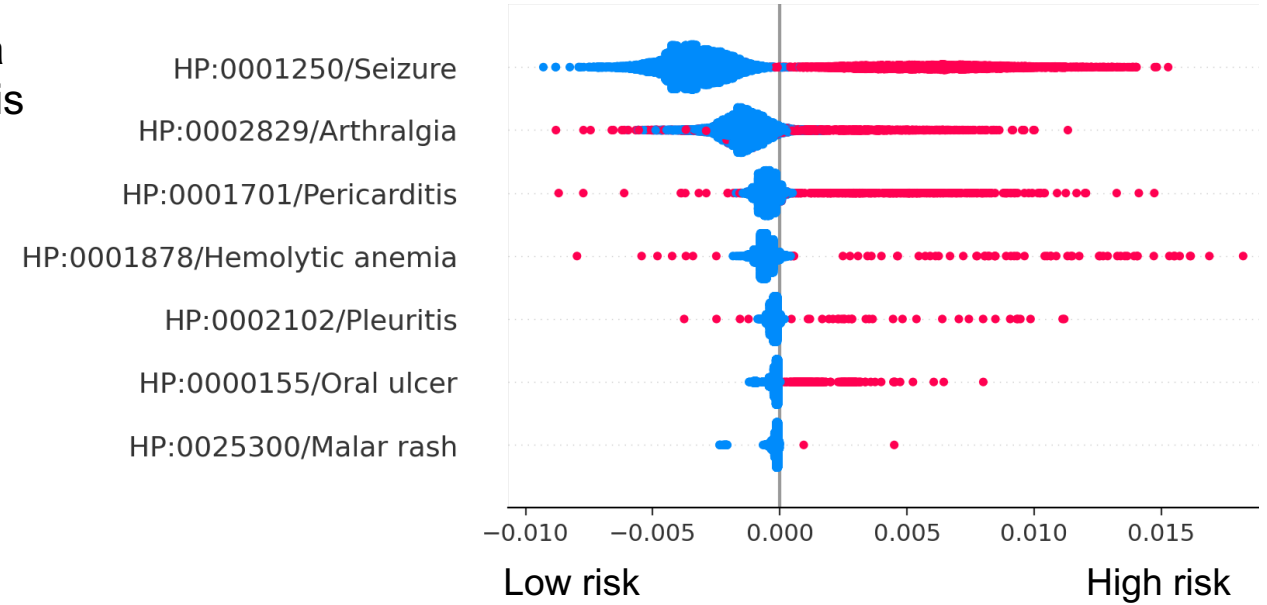


Imperial College London

Pangaea

Find the Patient at Risk of an Autoimmune Disease

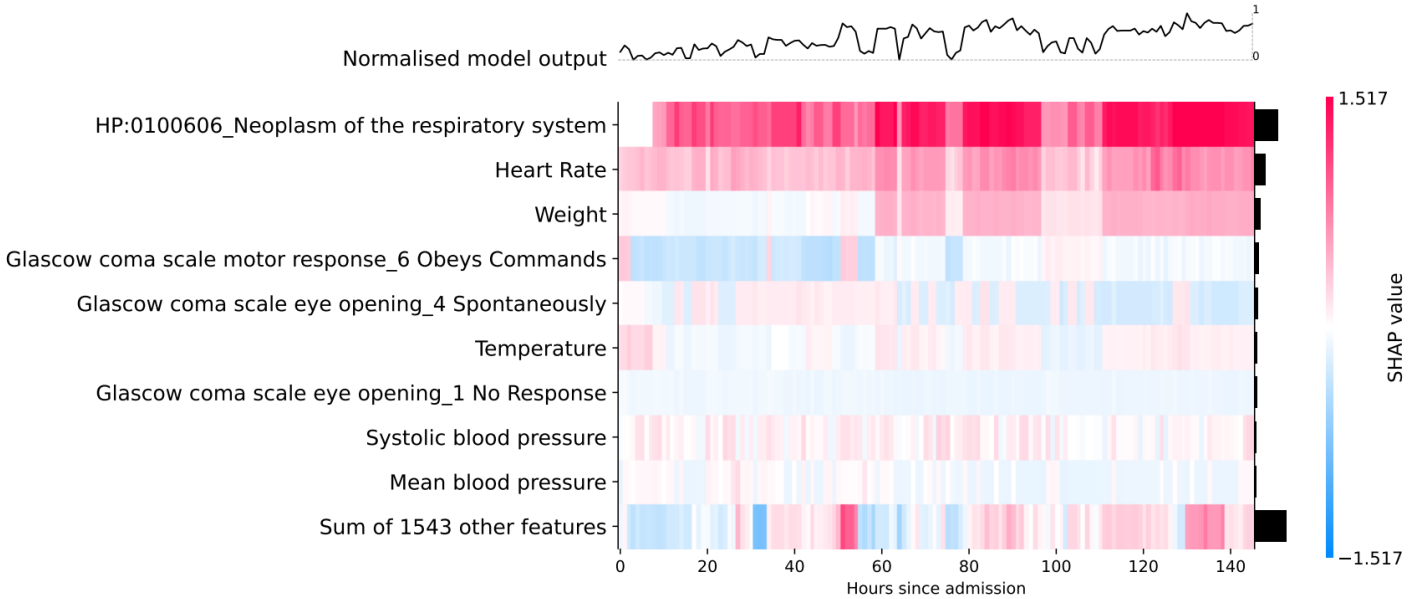
- Blue indicates absence of a feature e.g., no osteoporosis
- Red indicates feature presence



Imperial College London

Pangaea

Find the Patient at Risk of Death in ICU

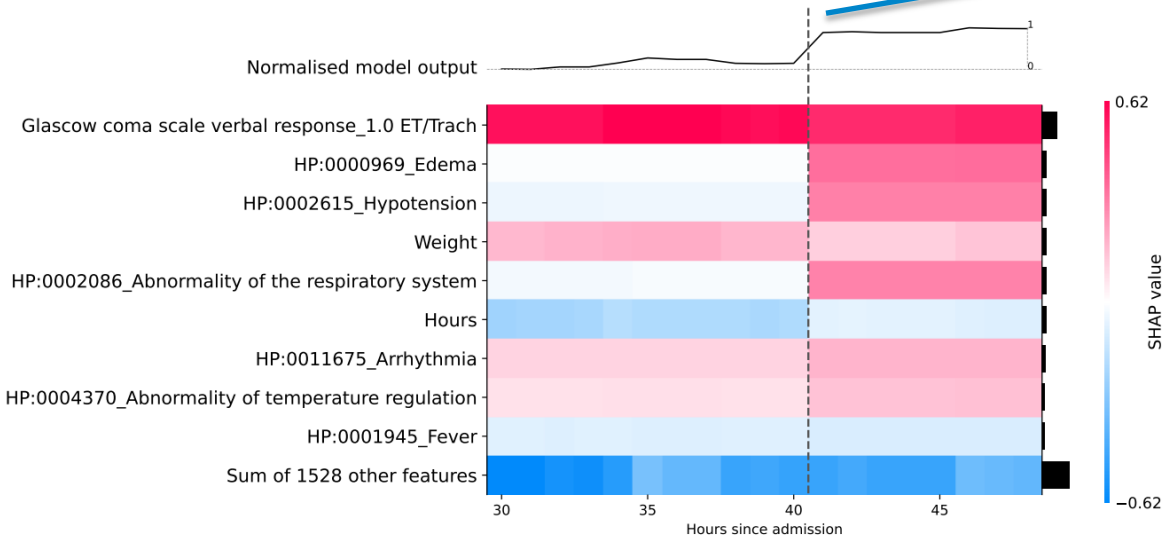


Imperial College London

Pangaea

Understand Disease Progression

When first note available



ICU length-of-stay

Imperial College London

Pangaea

Other Applications of Clinical Natural Language Processing

- Personalised medication / treatment
- Clinical trial recruitment and drug development
- Adverse drug events

Imperial College London

Pangaea

Data Privacy

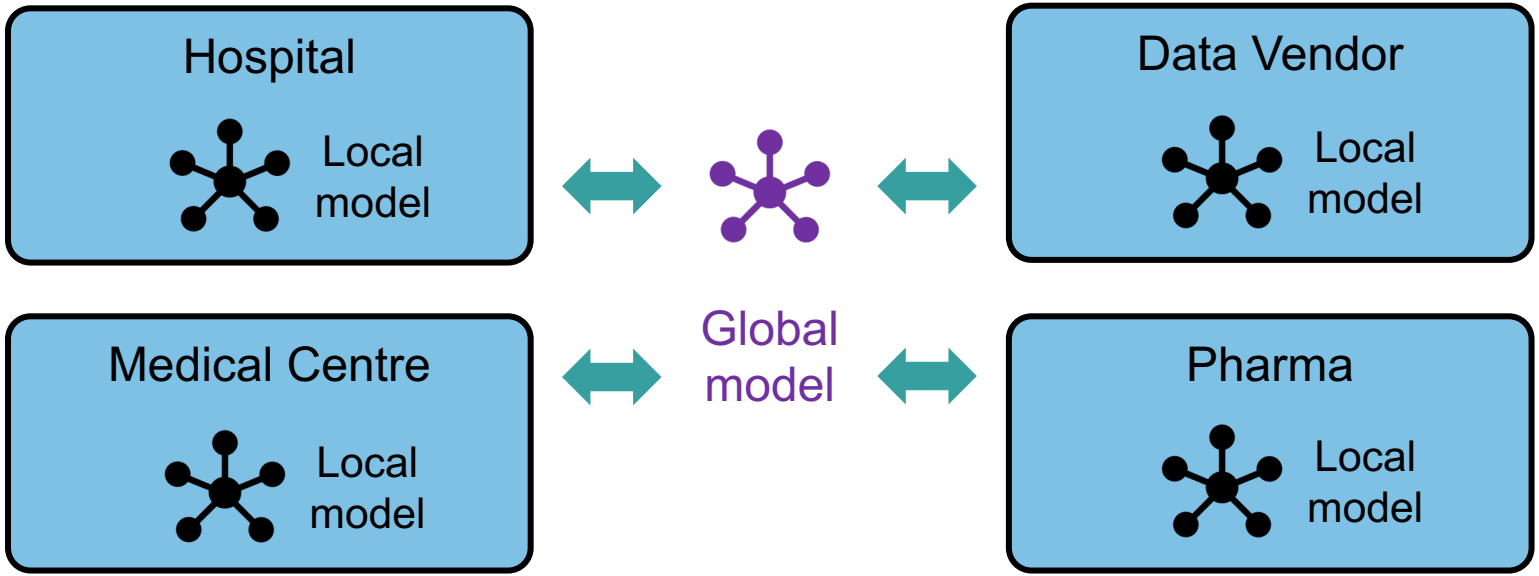
- Data privacy is very important!
- Transmitting identifiable data via Internet typically is not allowed.
- Data in a local data centre may not be sufficient especially rare diseases.

- How to build a model across different data centres without moving data?
- How to build a model if real data is limited?

Imperial College London

Pangaea

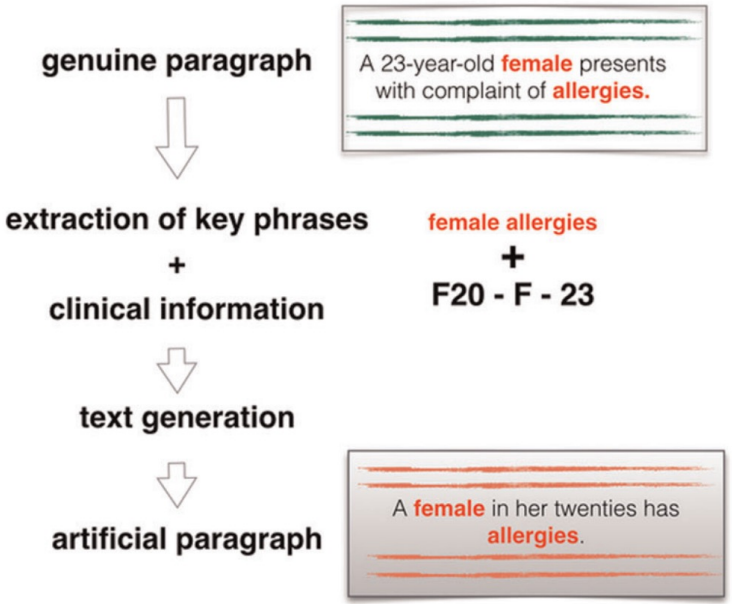
Federated Learning with NLP



Imperial College London

Pangaea

Synthetic Data Generation

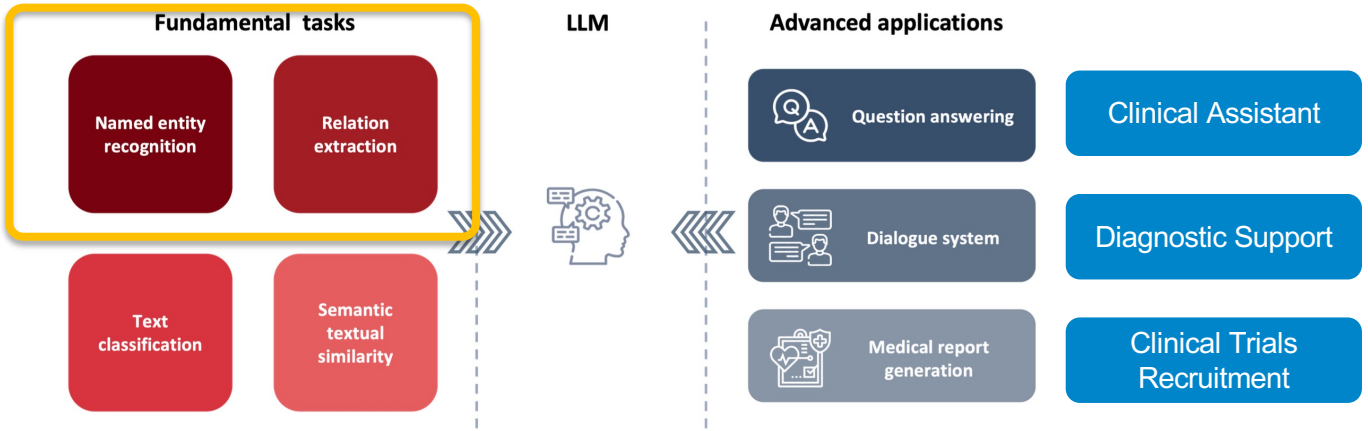


Imperial College London

Pangaea

Large Language Models to Assist Doctors on Tasks

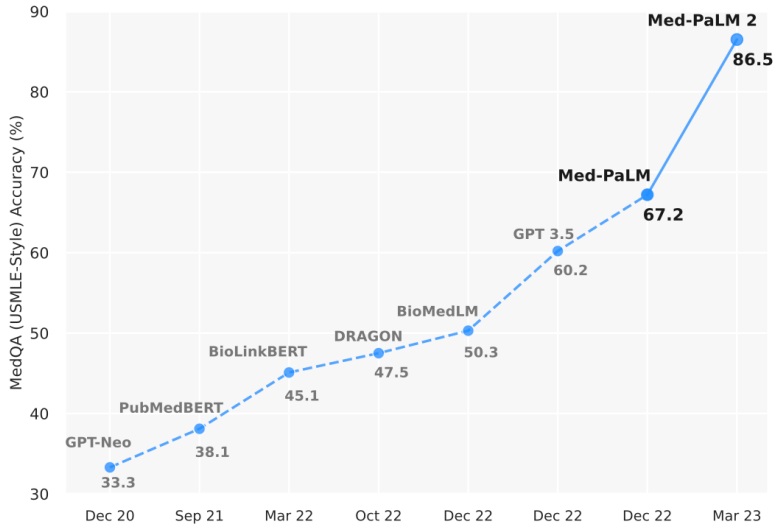
Reasoning with numbers

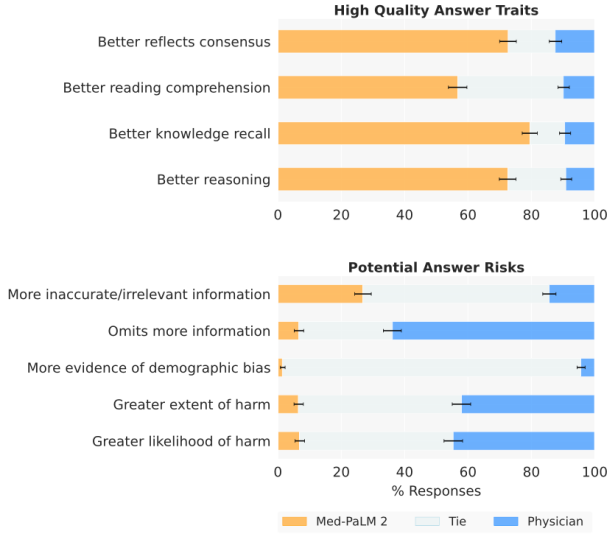


Imperial College London

Pangaea

Large Language Models for Medical Question Answering





Imperial College London

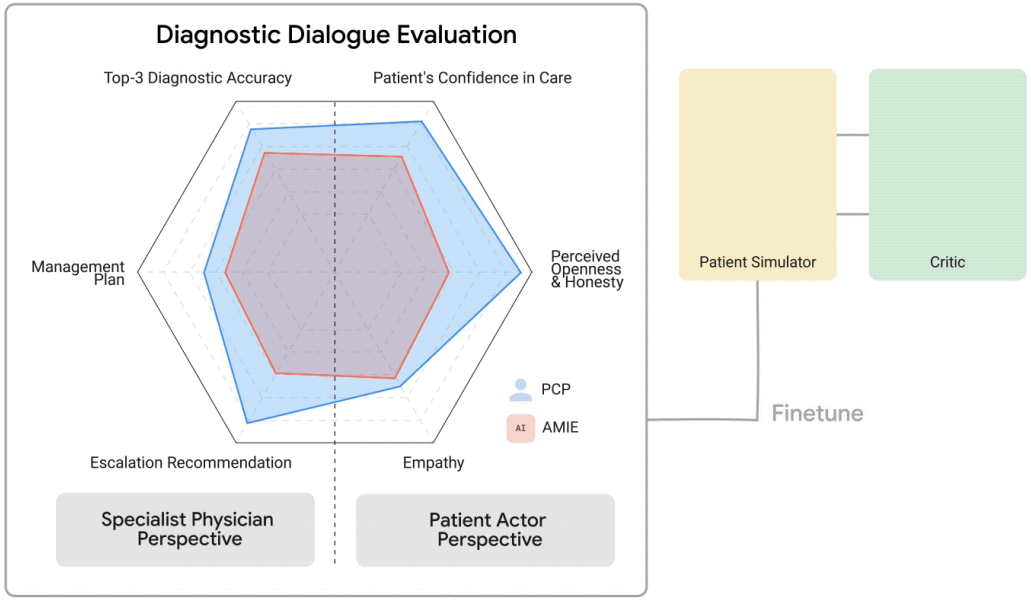
Pangaea

Large Language Models for Diagnostic Conversation

Imperial College London

Pangaea

Large Language Models for Diagnostic Conversation



Imperial College London

Pangaea

Thank you

1. Zhang, Jingqing. "Language modelling for clinical natural language understanding and generation." (2022).
2. Bommasani, Rishi, et al. "On the opportunities and risks of foundation models." *arXiv preprint arXiv:2108.07258* (2021).
3. Manning, Christopher, and Richard Socher. "Natural Language Processing with Deep Learning CS224N/Ling284." (2019).
4. Ian Goodfellow, Yoshua Bengio, and Aaron Courville. Deep learning. MIT press, 2016.
5. Mikolov, Tomas, et al. "Efficient estimation of word representations in vector space." *arXiv preprint arXiv:1301.3781* (2013).
6. Pennington, Jeffrey, Richard Socher, and Christopher D. Manning. "Glove: Global vectors for word representation." Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). 2014.
7. Liu, Frederick, et al. "Learning character-level compositionality with visual features." arXiv preprint arXiv:1704.04859 (2017).
8. Wu, Wei, et al. "Glyce: Glyph-vectors for Chinese Character Representations." arXiv preprint arXiv:1901.10125 (2019).
9. Devlin, Jacob, et al. "Bert: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805 (2018).
10. PEGASUS: A State-of-the-Art Model for Abstractive Text Summarization https://ai.googleblog.com/2020/06/pegasus-state-of-art-model-for.html
11. Zhang, Jingqing, et al. “Pegasus: Pre-training with extracted gap-sentences for abstractive summarization.” ICML 2020.
12. Vaswani, Ashish, et al. "Attention is all you need." *Advances in neural information processing systems* 30 (2017).
13. https://magazine.sebastianraschka.com/p/understanding-large-language-models
14. Yang, Jingfeng, et al. "Harnessing the power of llms in practice: A survey on chatgpt and beyond." *arXiv preprint arXiv:2304.13712* (2023).
15. Tanwar, Ashwani, et al. "Phenotyping in clinical text with unsupervised numerical reasoning for patient stratification." *Experimental Biology and Medicine* 247.22 (2022): 2038-2052.
16. Zhang, Jingqing, et al. "Clinical utility of automatic phenotype annotation in unstructured clinical notes: intensive care unit use." BMJ Health & Care Informatics 29.1 (2022).
17. Ive, Julia, et al. "Generation and evaluation of artificial mental health records for Natural Language Processing." *NPJ digital medicine* 3.1 (2020): 1-9.
18. Zhang, Jingqing, et al. "The Potential and Pitfalls of using a Large Language Model such as ChatGPT or GPT-4 as a Clinical Assistant." *arXiv preprint arXiv:2307.08152* (2023).
19. Tu, Tao, et al. "Towards Conversational Diagnostic AI." arXiv preprint arXiv:2401.05654 (2024).
20. He, Kai, et al. "A survey of large language models for healthcare: from data, technology, and applications to accountability and ethics." arXiv preprint arXiv:2310.05694 (2023).
21. Singhal, Karan, et al. "Towards expert-level medical question answering with large language models." arXiv preprint arXiv:2305.09617 (2023).