Natural Language Processing with Machine Learning

Yogesh Haribhau Kulkarni

YHK

Outline

About Me

YHK

Yogesh Haribhau Kulkarni

Bio:

- 20+ years in CAD/Engineering software development
- Got Bachelors, Masters and Doctoral degrees in Mechanical Engineering (specialization: Geometric Modeling Algorithms).
- Currently doing Coaching in fields such as Data Science, Artificial Intelligence Machine-Deep Learning (ML/DL) and Natural Language Processing (NLP).
- Feel free to follow me at:
  - Github (github.com/yogeshhk)
  - LinkedIn (www.linkedin.com/in/yogeshkulkarni/)
  - Medium (yogeshharibhaukulkarni.medium.com)
  - Send email to yogeshkulkarni at yahoo dot com

Office Hours: Saturdays, 2 to 5pm (IST); Free-Open to all; email for appointment.

YHK

Overview of NLP with Embeding

YHK

Use case: Bank Call Center

Calling the Call Center

- Calling to an IVR (Integrated voice response)
- A prerecorded menu selection.
- "Please press 1 for Account Details, Please press 2 for . . . "
- till it comes to your option.
- towards end, somewhere, given access to a person to talk to.

Boring? Annoying?

(Ref: Deep Learning and NLP A-Z - Kirill Eremenko)

Instead, how about typing/saying your query directly and getting the answer right away?

YHK

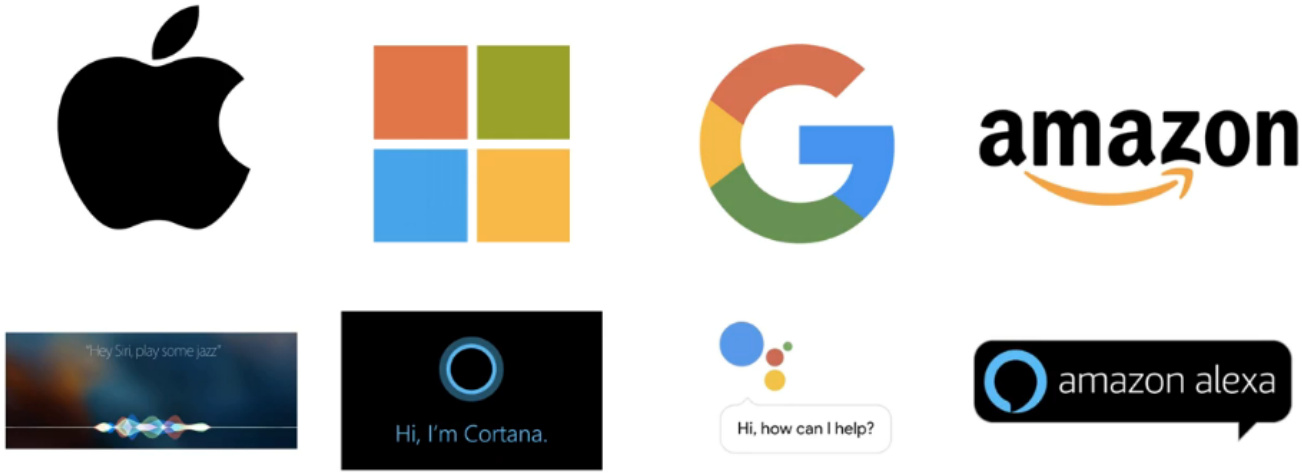
Solution

Chatbots

- Which problem of IVR it is solving?
- Advantages?
- Disadvantages?
- Gaining popularity . . .
- Many platforms
- Companies in Pune?

YHK

The Giants are at it ...



(Ref: Deep Learning and NLP A-Z - Kirill Eremenko)

- Chatbots or QA systems, predominantly voice based,
- Underlying processing is primarily Natural Language Processing (NLP).
- You can have your own chatbot, specific to you!!
- NLP is the core skill needed.

YHK

Why so much popularity?

Chatbots are:

- Autonomous and Always Available
- Drive Conversation
- Able to handle millions of requests, scalable.

Its hard to master language, and thus NLP.

YHK

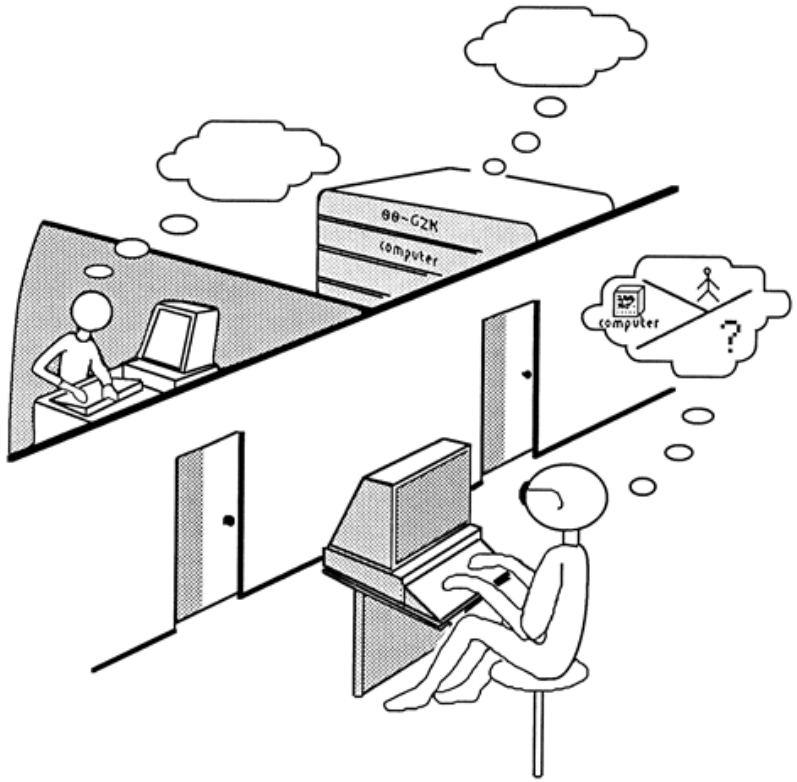
NLP is AI-complete

- "The most difficult problems in AI manifest themselves in human language phenomena."
- Use of language is the touchstone of intelligent behavior.
- Test for Intelligence - Turing Test
- Alan Turing (1950) proposed a test of a machine's capability to perform human-like conversation.

YHK

Turing Test

A human judge engages in a natural language conversation with two other parties, one a human and the other a machine; if the judge cannot reliably tell which is which, then the machine is said to pass the test.



YHK

Early Conversational Programs

- ELIZA (by Joseph Weizenbaum), 1966.
- A psychotherapist, but NO real understanding;
- Simple pattern-matching to respond to user input to canned responses

(my ?x depresses me) (why does your ?x depress you)
(life ?x) (why do you say it ?x)
(I could ?x) (you could ?x)
(because ?x) (that is a good reason)
(?x) (tell me more)

```
>> Hello.
How do you do.  Please state your problem.
>> Men are all alike.
In what way ?
>> They're alwas bugging us about something or other.
I'm not sure I understand you fully.
>> They're always bugging us about something or other.
Can you think of a specific example ?
>> Well, my boyfriend made me come here.
Your boyfriend made you come here  ?
>> He says I'm depressed much of the time.
I am sorry to hear that you are depressed.
```

YHK

Loebner Prize

- In 1990, Hugh Loebner started Turing Test competition
- $100,000 will be awarded to the first bot that judges cannot distinguish from a real human in a Turing test that includes text, visual, and auditory input.
- Nobody has won the grand prize yet.
- 2016 (and 2013) year-wise top winner - Mitsuku.
  https://www.facebook.com/mitsukubot

Why can't we win the Grand Prize? What are the challenges? Why Language is hard? What is Language?

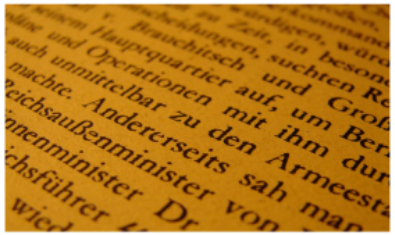
YHK

What is Language?

YHK

Language Types

Natural Language



(http://expertenough.com/2392/german-language-hacks)

日本語で
冬は世界各地でさまざまなお祝いが行われる時期です。ほんのいくつか例を挙げるだけでも、ハナカ、クリスマス、クワンザ、新年などさまざまなお祝いがあります。各文化によってその祝い方はさまざまですが、ほとんどのお祝いにはごちそうが欠かせません。

(http://www.transparent.com/learn-japanese/articles/dec_99.html)

Artificial Language

```
try {
    cMessage = messageQueue.take();
    for (AsyncContext ac : queue) {
        try {
            PrintWriter acWriter = ac.g
            acWriter.println(cMessage);
            acWriter.flush();
        } catch(IO
            System
            queue.
        }
    }
} catch(Interrupte
```

append(CharSequenc
append(char c)
append(CharSequenc
format(String forma
format(Locale l, S
printf(String forma

(https://netbeans.org/features/java/)

```
def add5(x):
    return x+5

def dotwrite(ast):
    nodename = getNodename()
    label=symbol.sym_name.get(int(ast[0]),ast[0])
    print '    %s [label="%s" % (nodename,label),
    if isinstance(ast[1], str):
        if ast[1].strip():
            print '= %s"];' % ast[1]
        else:
            print '"]'
    else:
        print '"];'
        children = []
        for n, child in enumerate(ast[1:]):
            children.append(dotwrite(child))
        print '    %s -> {' % nodename,
        for name in children:
            print '%s' % name,
```

(http://noobite.com/learn-programming-start-with-python/)

Differences?

YHK

Language, simplistically

- A vocabulary consists of a set of words
- A text is composed of a sequence of words from a vocabulary
- A language is constructed of a set of all possible texts

(http://learnenglish.britishcouncil.org/en/vocabulary-games)

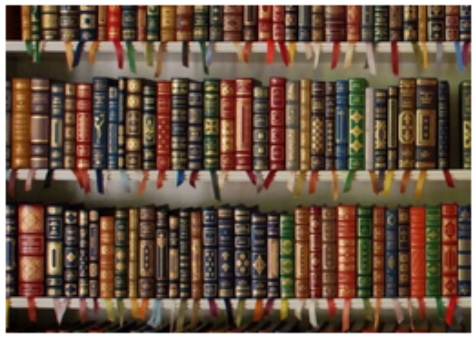

THIS WEEK

EDITORIALS

Beyond the genome

Studies of the epigenomic signatures of many healthy and diseased human tissues could provide crucial information to link genetic variation and disease.

(http://www.nature.com/polopoly_fs/1.16929!/menu/main/topColumns/topLeftColumn/pdf/518273a.pdf)



(http://www.old-engli.sh/language.php)

YHK

NLP

- NLP is Natural Language Processing, ie processing Natural Langauge for some end-purpose in mind.
- Inspite of usage of Natural Language for thousands of years, why are we not able to process it well?

YHK

NLP Challenges

Paraphrasing

Paraphrasing: Different words/sentences express the same meaning

- Season of the year: Fall/Autumn
- Book delivery time
  - When will my book arrive?
  - When will I receive my book?

YHK

Ambiguity

Ambiguity: One word/sentence can have different meanings

- Fall
  - The third season of the year
  - Moving down towards the ground or towards a lower position
- The door is open
  - Expressing a fact
  - A request to close the door

YHK

Syntax and ambiguity

"I saw the man with a telescope."
- Who had the telescope?

YHK

Semantics

The astronomer loves the star.

- Star in the sky
- Celebrity

(http://en.wikipedia.org/wiki/Star#/media/File:Starsinthesky.jpg)

(http://www.businessnewsdaily.com/2023-celebrity-hiring.html)

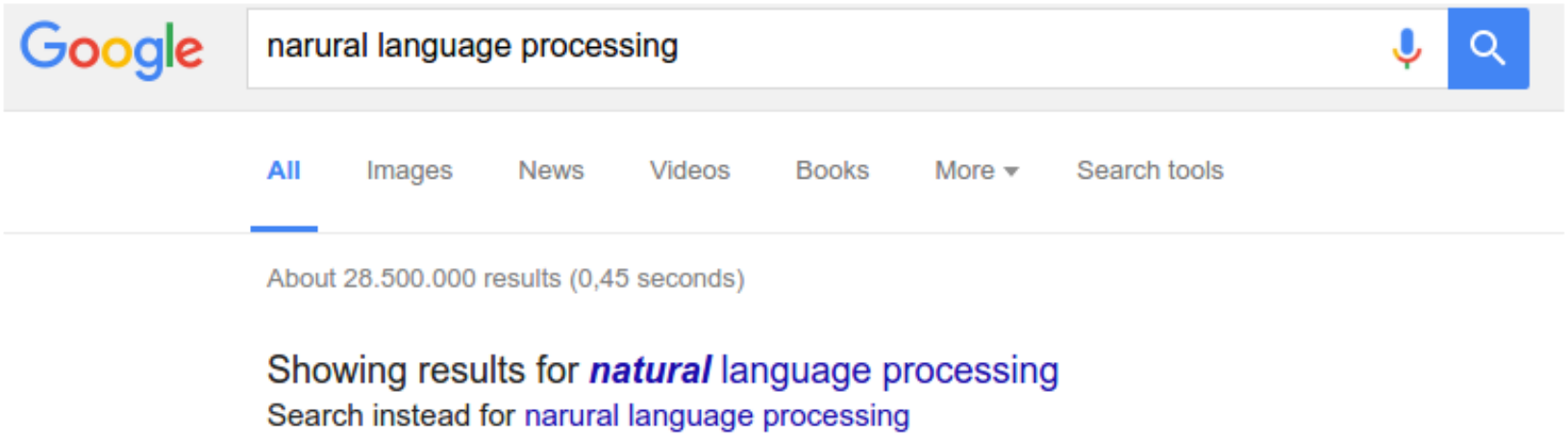
YHK

NLP Applications

YHK

Spell and Grammar Checking

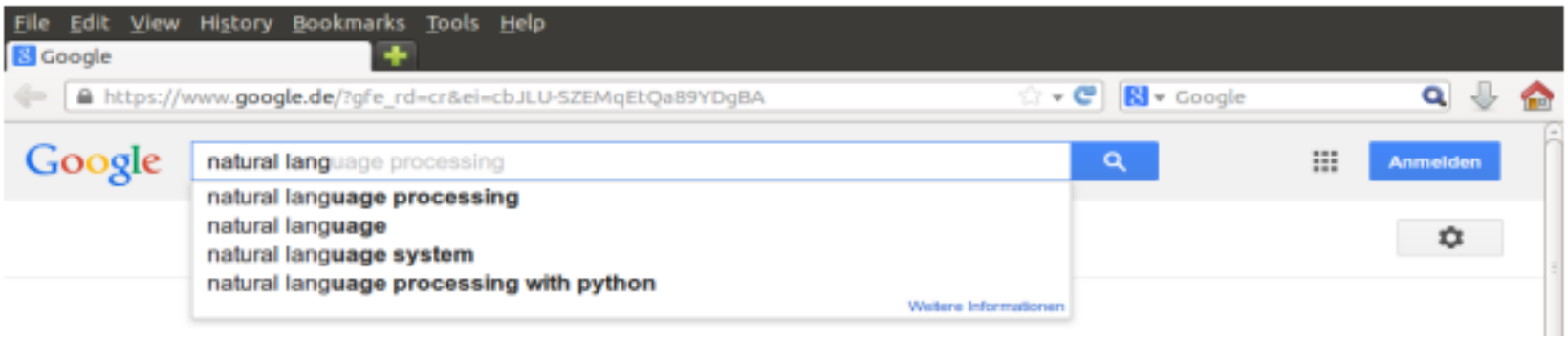
- Checking spelling and grammar
- Suggesting alternatives for the errors



Word Prediction

Word Prediction: Predicting the next word that is highly probable to be typed by the user

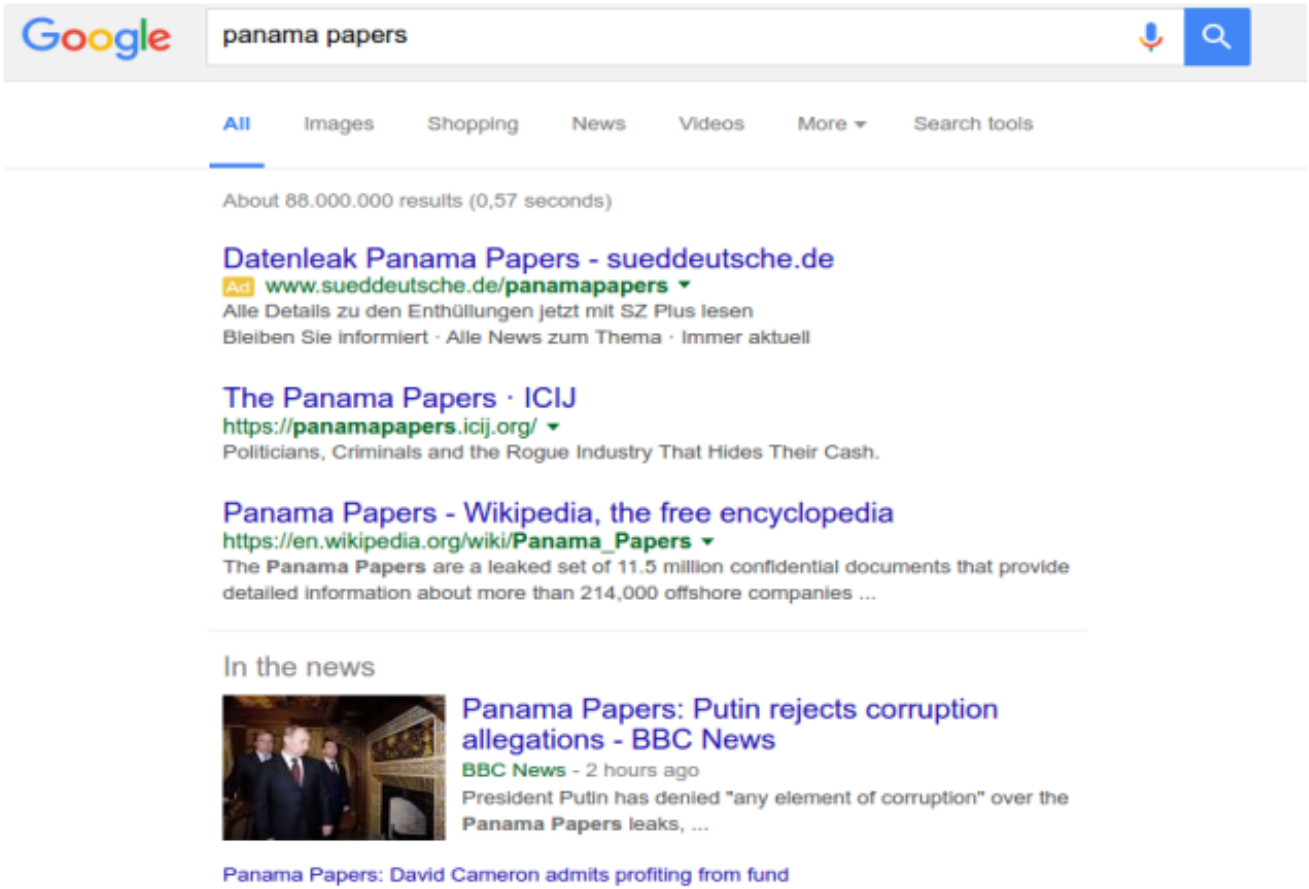
- Mobile typing
- Search Engines



YHK

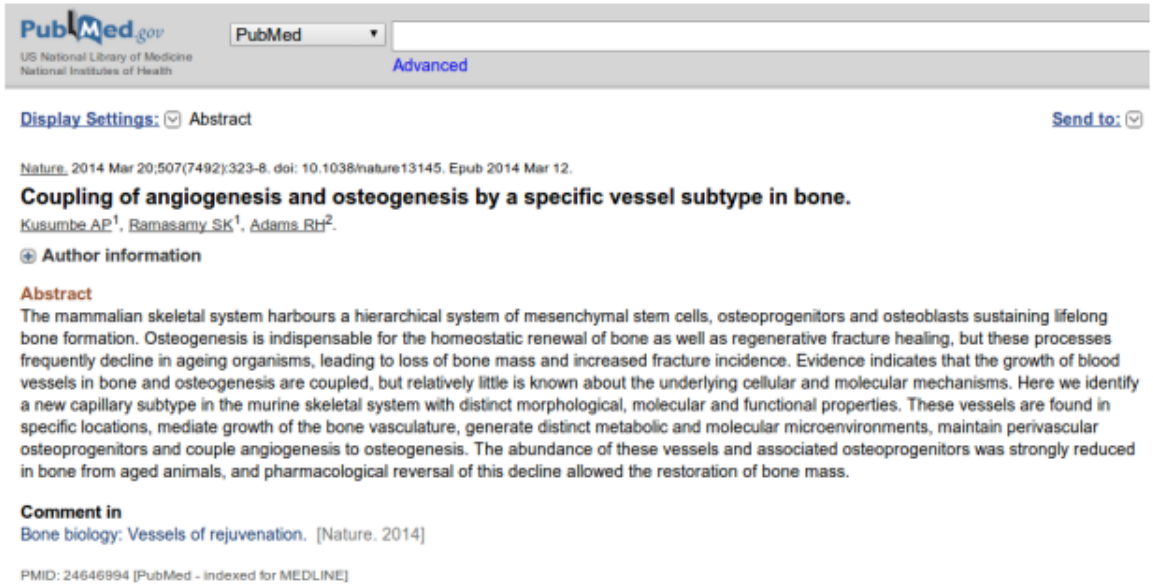
Information Retrieval

Information Retrieval: Finding relevant information to the user's query



Text Categorization

Text Categorization: Assigning one (or more) pre-defined category to a text



PubMed.gov
US National Library of Medicine
National Institutes of Health

PubMed

Advanced

Display Settings: Abstract

Send to:

Nature. 2014 Mar 20;507(7492):323-8. doi: 10.1038/nature13145. Epub 2014 Mar 12.

Coupling of angiogenesis and osteogenesis by a specific vessel subtype in bone.

Kusumbe AP[1], Ramasamy SK[1], Adams RH[2].

Author information

Abstract

The mammalian skeletal system harbours a hierarchical system of mesenchymal stem cells, osteoprogenitors and osteoblasts sustaining lifelong bone formation. Osteogenesis is indispensable for the homeostatic renewal of bone as well as regenerative fracture healing, but these processes frequently decline in ageing organisms, leading to loss of bone mass and increased fracture incidence. Evidence indicates that the growth of blood vessels in bone and osteogenesis are coupled, but relatively little is known about the underlying cellular and molecular mechanisms. Here we identify a new capillary subtype in the murine skeletal system with distinct morphological, molecular and functional properties. These vessels are found in specific locations, mediate growth of the bone vasculature, generate distinct metabolic and molecular microenvironments, maintain perivascular osteoprogenitors and couple angiogenesis to osteogenesis. The abundance of these vessels and associated osteoprogenitors was strongly reduced in bone from aged animals, and pharmacological reversal of this decline allowed the restoration of bone mass.

Comment in

Bone biology: Vessels of rejuvenation. [Nature. 2014]

PMID: 24646994 [PubMed - indexed for MEDLINE]

MeSH Terms

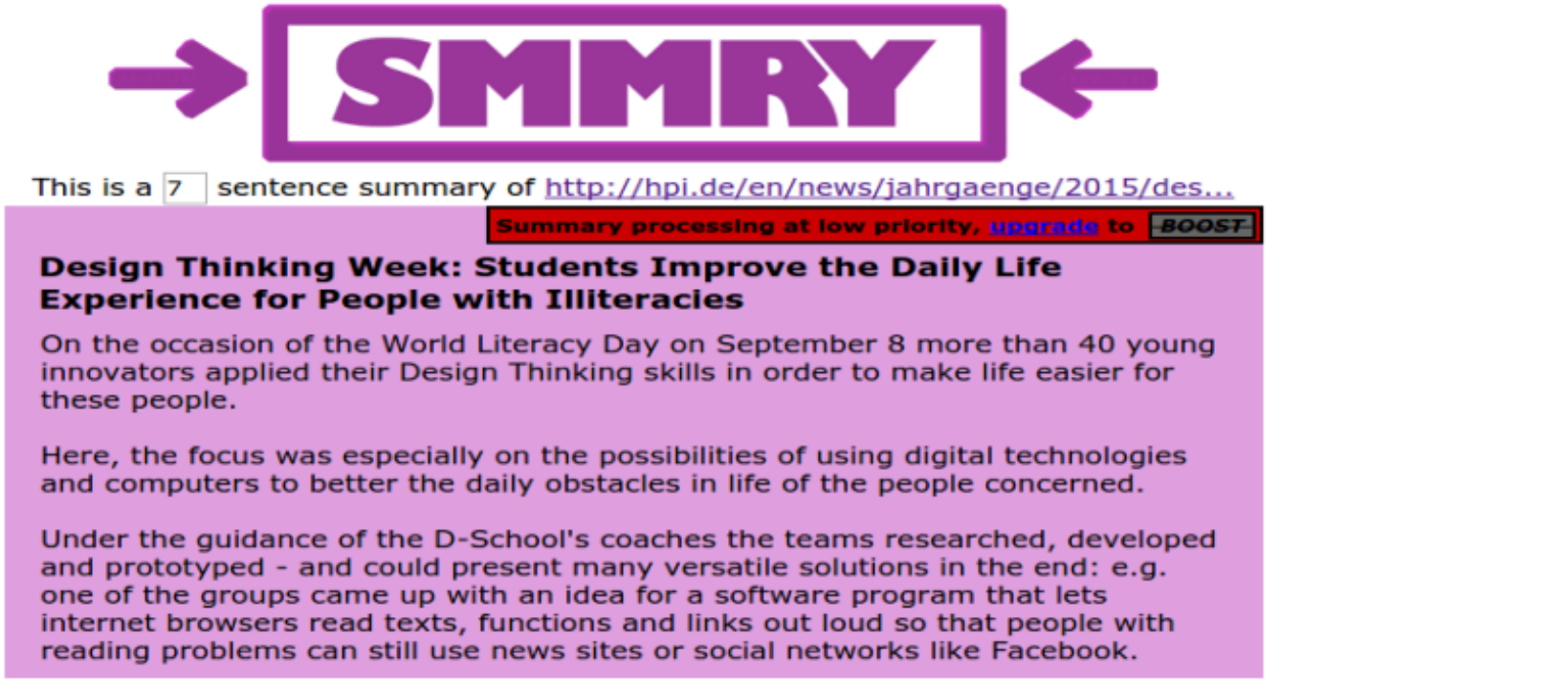
Aging/metabolism
Aging/pathology
Animals
Blood Vessels/anatomy & histology
Blood Vessels/cytology
Blood Vessels/growth & development
Blood Vessels/physiology*
Bone and Bones/blood supply*
Bone and Bones/cytology
Endothelial Cells/metabolism
Hypoxia-Inducible Factor 1, alpha Subunit/metabolism
Male
Mice
Mice, Inbred C57BL
Neovascularization, Physiologic/physiology*
Osteoblasts/cytology
Osteoblasts/metabolism
Osteogenesis/physiology*
Oxygen/metabolism
Stem Cells/cytology
Stem Cells/metabolism

YHK

Text Categorization

Summarization

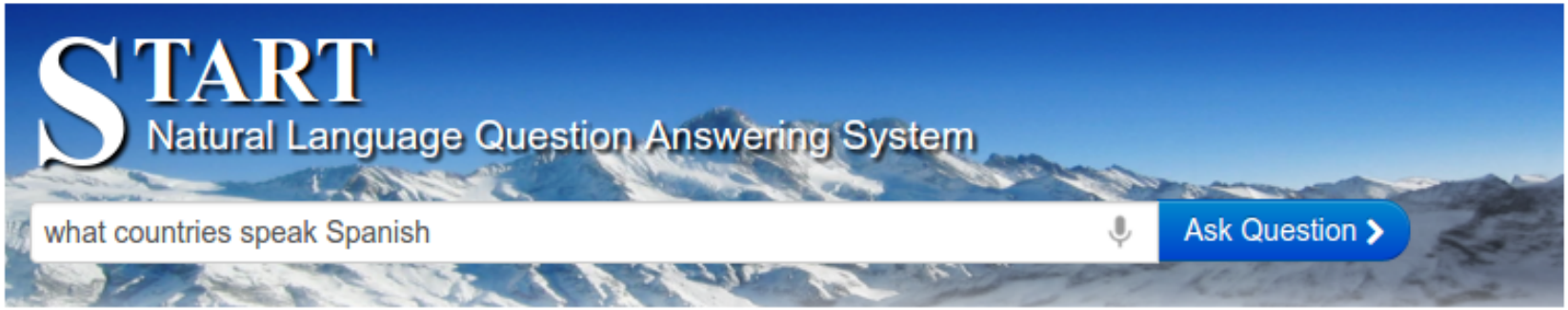
Summarization: Generating a short summary from one or more documents, sometimes based on a given query



http://smmry.com/

Question answering

Question answering: Answering questions with a short answer



===> what countries speak Spanish

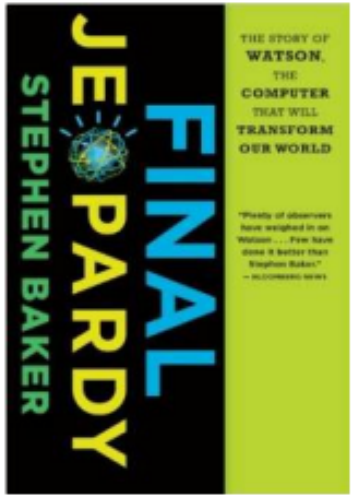
The language Spanish is spoken in Argentina, Aruba, Belize, Bolivia, Brazil, Canada, Cayman Islands, Chile, Colombia, Costa Rica, Cuba, Curacao, Dominican Republic, Ecuador, El Salvador, Equatorial Guinea, Falkland Islands (Islas Malvinas), Gibraltar, Guatemala, Honduras, Mexico, Nicaragua, Panama, Paraguay, Peru, Puerto Rico, Saint Martin, Sint Maarten, Spain, Switzerland, Trinidad and Tobago, United States, Uruguay, Venezuela, and Virgin Islands.

The language Castilian Spanish is spoken in Spain.

YHK

Question answering

Question answering: IBM Watson in Jeopardy



YHK

Information Extraction

Information Extraction: Extracting important concepts from texts and assigning them to slot in a certain template

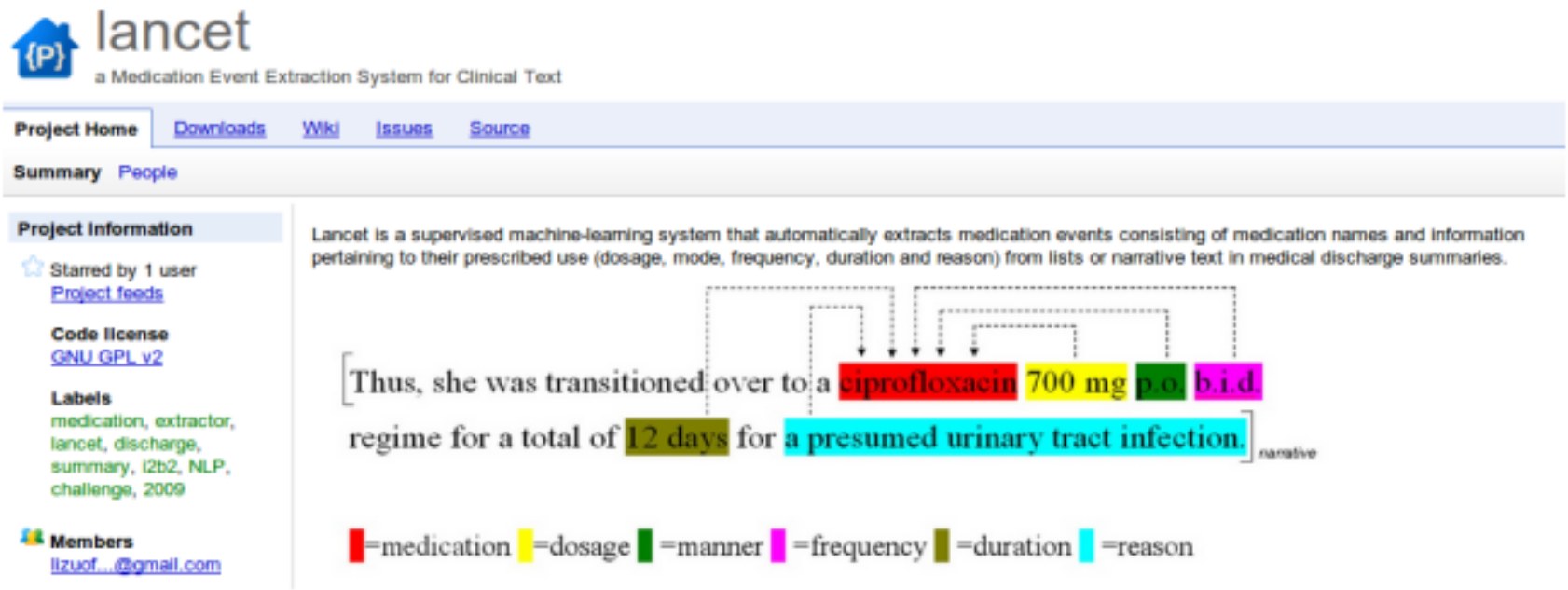
Merkel at the EPP Summit, March 2016

Chancellor of Germany	
Incumbent	
Assumed office	22 November 2005
President	Horst Köhler Christian Wulff Joachim Gauck
Deputy	Franz Müntefering Frank-Walter Steinmeier Guido Westerwelle Philipp Rösler Sigmar Gabriel
Preceded by	Gerhard Schröder
Leader of the Christian Democratic Union	
Incumbent	
Assumed office	

In office	17 November 1994 – 26 October 1998
Chancellor	Helmut Kohl
Preceded by	Klaus Töpfer
Succeeded by	Jürgen Trittin
Minister for Women and Youth	
In office	18 January 1991 – 17 November 1994
Chancellor	Helmut Kohl
Preceded by	Ursula Lehr
Succeeded by	Claudia Nolte
Personal details	
Born	Angela Dorothea Kasner 17 July 1954 (age 61) Hamburg, West Germany
Political party	Democratic Awakening (1989–1990) Christian Democratic Union (1990–present)
Spouse(s)	Ulrich Merkel (1977–1982) Joachim Sauer (1998–present)
Alma mater	Leipzig University
Religion	Lutheranism (within Evangelical Church)
Signature	

YHK

Information Extraction: Includes named-entity recognition



Machine Translation

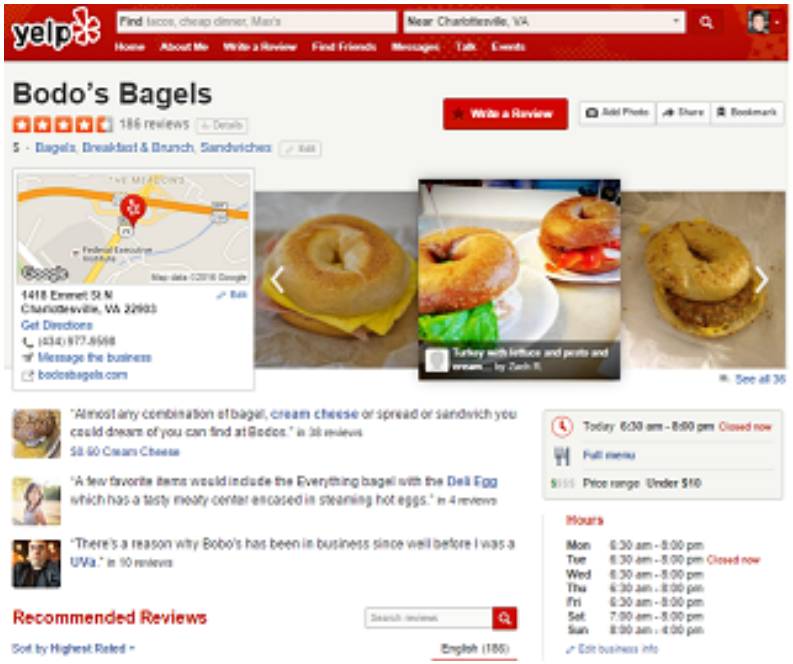
Machine Translation: Translating a text from one language to another

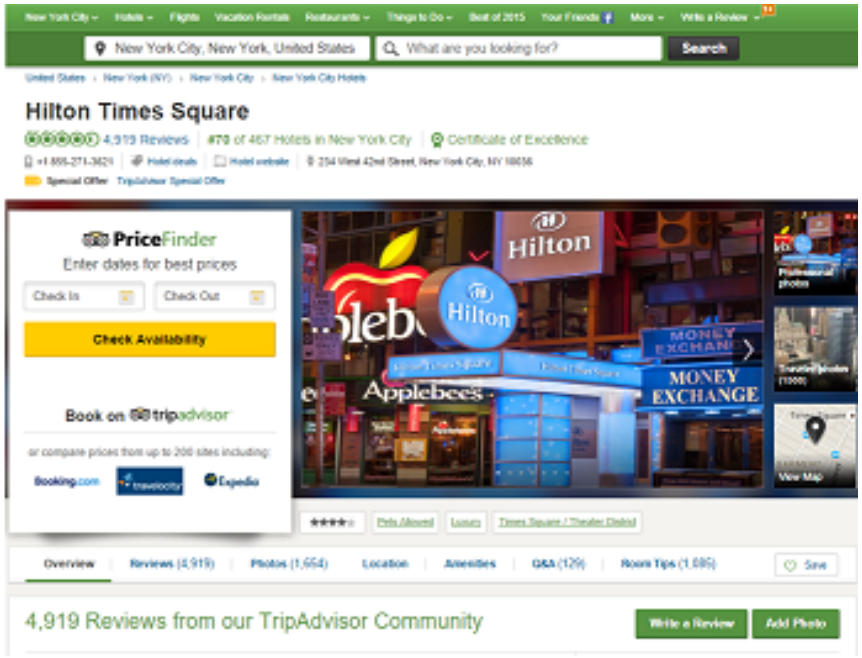
YHK

Sentiment Analysis

Sentiment Analysis: Identifying sentiments and opinions stated in a text

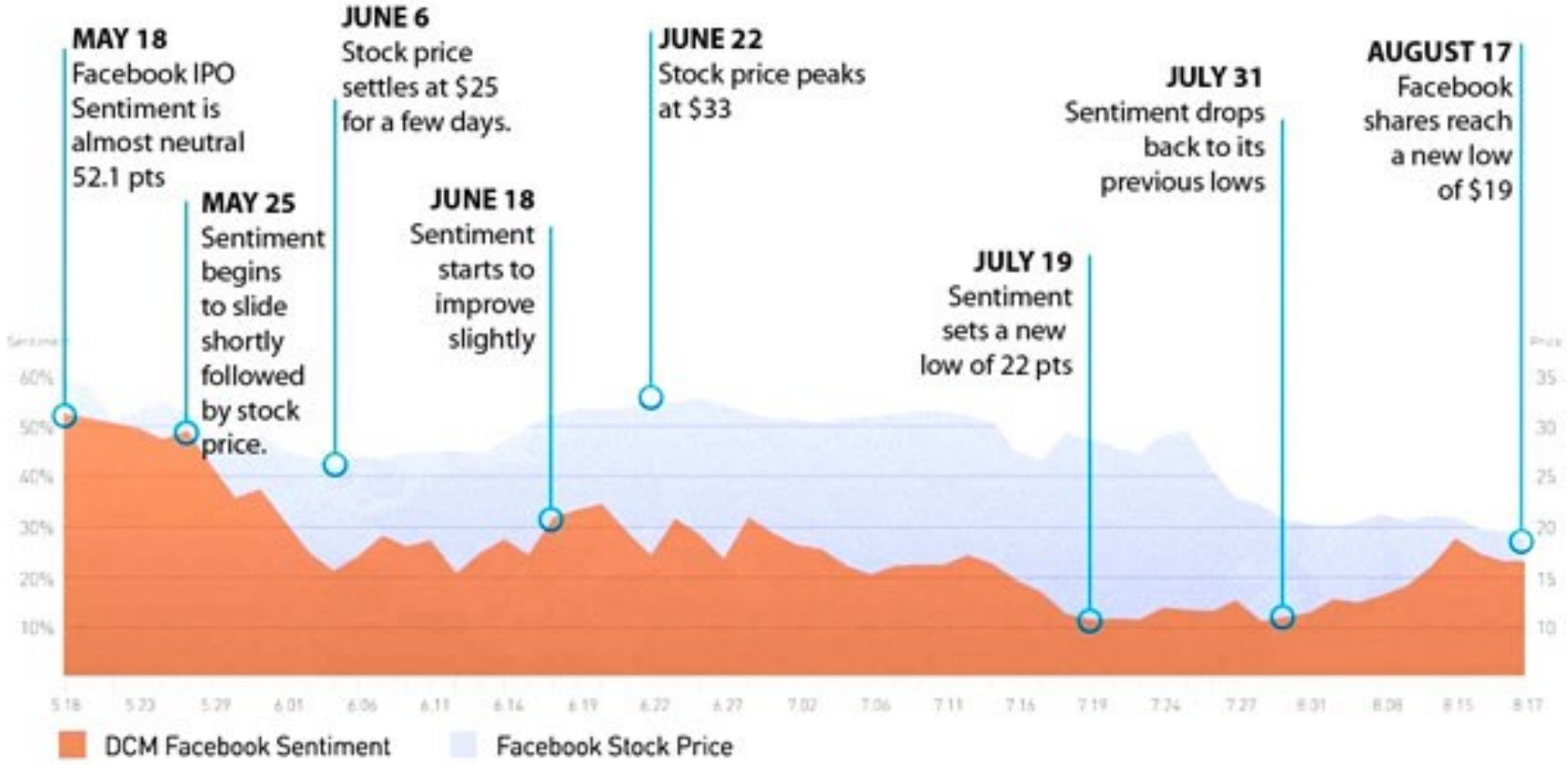
Restaurant/hotel recommendation, Product reviews





Sentiment Analysis

Text analytics in financial services



NLP in today's time

Trends:

- An enormous amount of information is now available in machine readable form as natural language text (newspapers, web pages, medical records, financial filings, product reviews, discussion forums, etc.)
- Conversational agents are becoming an important form of human-computer communication
- Much of human-human interaction is now mediated by computers via social media

Collectively, this means that copious data is available to be used in the development of NLP systems.

YHK

Level of difficulties

- Easy (mostly solved)
  - Spell and grammar checking
  - Some text categorization tasks
  - Some named-entity recognition tasks
- Intermediate (good progress)
  - Information retrieval
  - Sentiment analysis
  - Machine translation
  - Information extraction
- Difficult (still hard)
  - Question answering
  - Summarization
  - Dialog systems

YHK

Langauge Representation: How to make text computable?

YHK

Document Representation & Language Model

- How to represent a document?
- Make it computable
- How to infer the relationship among documents or identify the structure within a document?
- Knowledge discovery
- Language Model and N-Grams

YHK

Bag-of-Words representation

Term as the basis for vector space

- Doc1: Text mining is to identify useful information.
- Doc2: Useful information is mined from text.
- Doc3: Apple is delicious.

	text	information	identify	mining	mined	is	useful	to	from	apple	delicious
Doc1	1	1	1	1	0	1	1	1	0	0	0
Doc2	1	1	0	0	1	1	1	0	1	0	0
Doc3	0	0	0	0	0	1	0	0	0	1	1

YHK

What are Word Vectors/Embeddings?

- Word Embeddings are the texts converted into numbers
- There may be different numerical representations of same text.
- Many Machine Learning algorithms and almost all Deep Learning Architectures are incapable of processing strings or plain text in their raw form.
- They require numbers as inputs to perform any sort of job, be it classification, regression etc. in broad terms.
- So, for the computer to be able to "understand" a vector representation of a word is required.

YHK

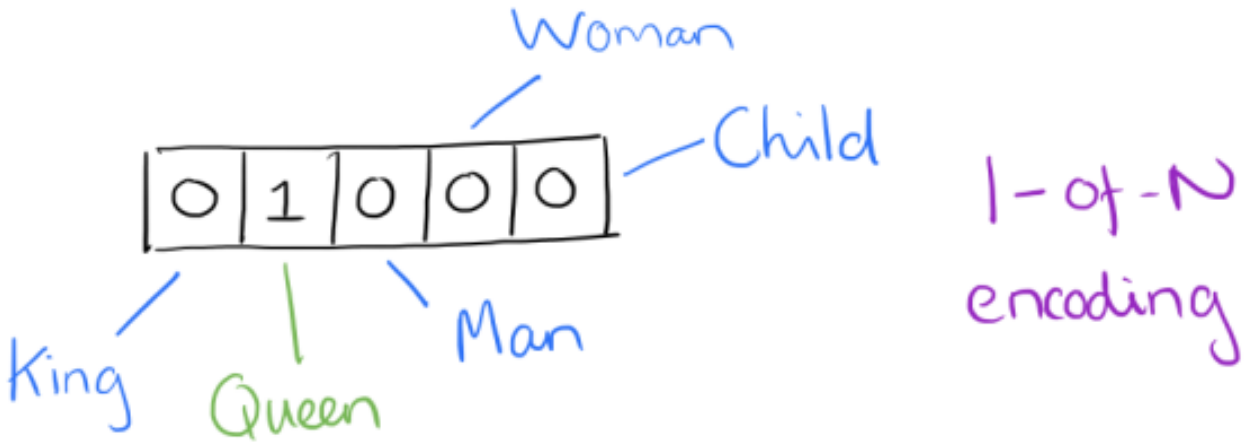
Different types of Word Vectors

- (Traditional) Frequency based Embedding:
  - One-hot
  - Count Vector
  - TF-IDF Vector
  - Co-Occurrence Vector
- (Modern) Prediction based Embedding:
  - Word2vec (Google)
  - Global Vector Representations (GloVe) (Stanford)

YHK

One Hot

One-hot: Suppose our vocabulary has only five words: King, Queen, Man, Woman, and Child. We could encode the word 'Queen' as:



No meaningful comparison possible.

YHK

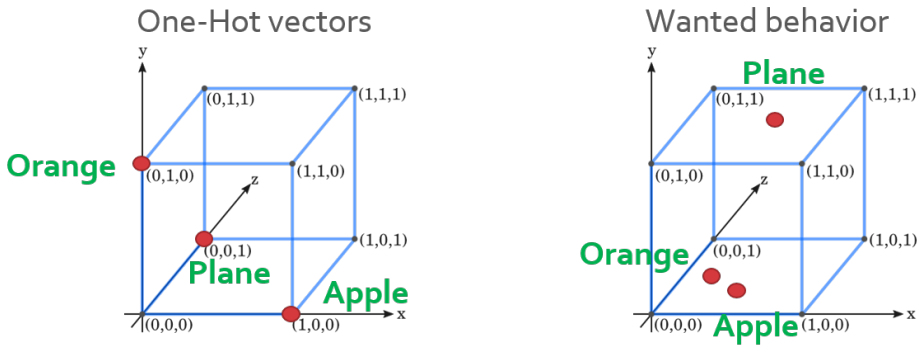
Good Vector Representation

- To have "Semantic" (meaning-wise) representation, the Similar words should be close to each other in the hyper dimensional space.
- Non-similar words should be far apart from each other in the hyper dimensional space.

YHK

Good Vector Representation

- Traditional One Hot Encoding:
  - Apple = [1, 0, 0]
  - Orange = [0, 1, 0]
  - Plane = [0, 0, 1]



- Very few cells participate in the representation.

Word2Vec

Word2vec (Google): a distributed representation of a word is used and not sparse like One-Hot.

Represent in some abstract way the 'meaning' of a word.

YHK

Word Distributed Representation - Word2Vec

- All vector cells participate in representing each word.
- Words are represented by real valued dense vectors of significantly smaller dimensions (e.g. 100 - 1000).
- Intuition: consider each vector cell as a representative of some feature.

Word Representations Comparison

Traditional Method - Bag of Words Model

- Uses one hot encoding
- Each word in the vocabulary is represented by one bit position in a HUGE vector.
- For example, with a vocabulary of 10000 words, and "Hello" is the 4th word in the dictionary: `0 0 0 1 0 0 . . . . . 0 0 0 0`
- Context information is not utilized

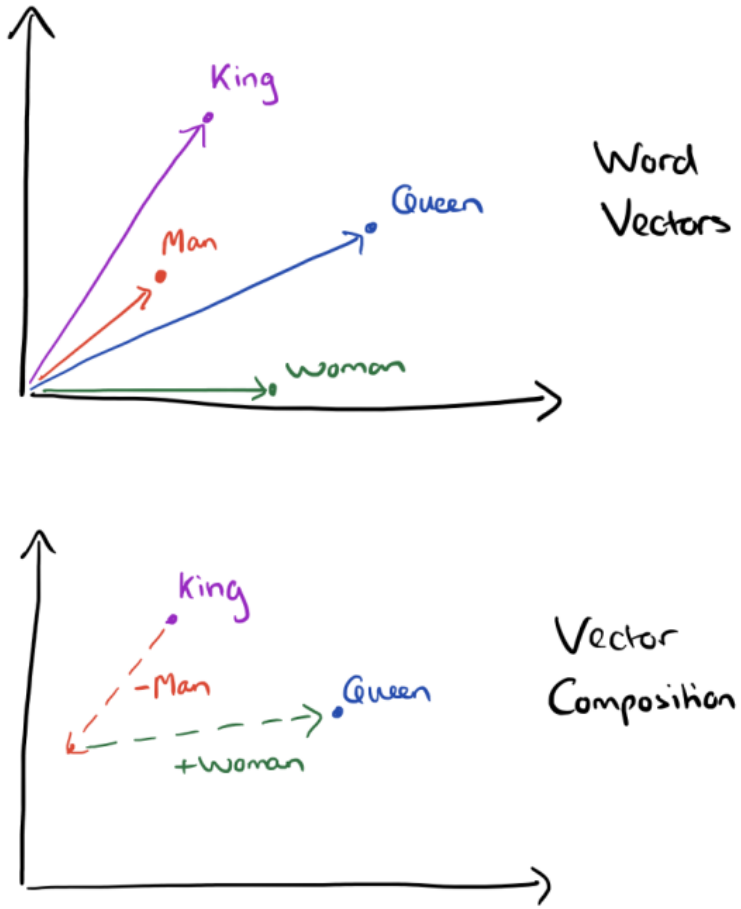
Modern - Word Vectors

- Stores each word in as a point in space, represented by a vector of fixed number of dimensions (generally 300)
- Unsupervised, built just by reading huge corpus
- For example, "Hello" might be represented as : `[0.4, -0.11, 0.55, 0.3 . . . 0.1, 0.02]`
- Context information is utilized

YHK

Examples

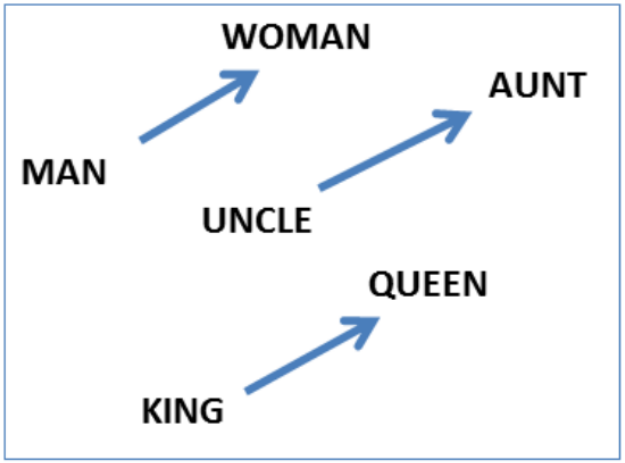
Vectors for King, Man, Queen, & Woman:



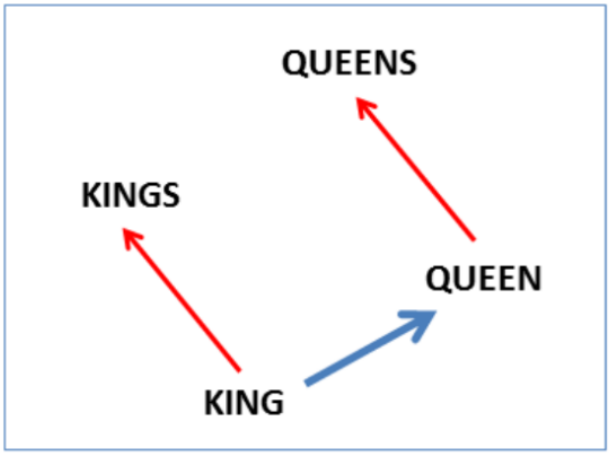
YHK

Examples

Gender relation:



Plural relation:



YHK

Examples

Word pair relationships:

Table 8: *Examples of the word pair relationships, using the best word vectors from Table 4 (Skip-gram model trained on 783M words with 300 dimensionality).*

Relationship	Example 1	Example 2	Example 3
France - Paris	Italy: Rome	Japan: Tokyo	Florida: Tallahassee
big - bigger	small: larger	cold: colder	quick: quicker
Miami - Florida	Baltimore: Maryland	Dallas: Texas	Kona: Hawaii
Einstein - scientist	Messi: midfielder	Mozart: violinist	Picasso: painter
Sarkozy - France	Berlusconi: Italy	Merkel: Germany	Koizumi: Japan
copper - Cu	zinc: Zn	gold: Au	uranium: plutonium
Berlusconi - Silvio	Sarkozy: Nicolas	Putin: Medvedev	Obama: Barack
Microsoft - Windows	Google: Android	IBM: Linux	Apple: iPhone
Microsoft - Ballmer	Google: Yahoo	IBM: McNealy	Apple: Jobs
Japan - sushi	Germany: bratwurst	France: tapas	USA: pizza

YHK

Examples

Country-capital city relationship:

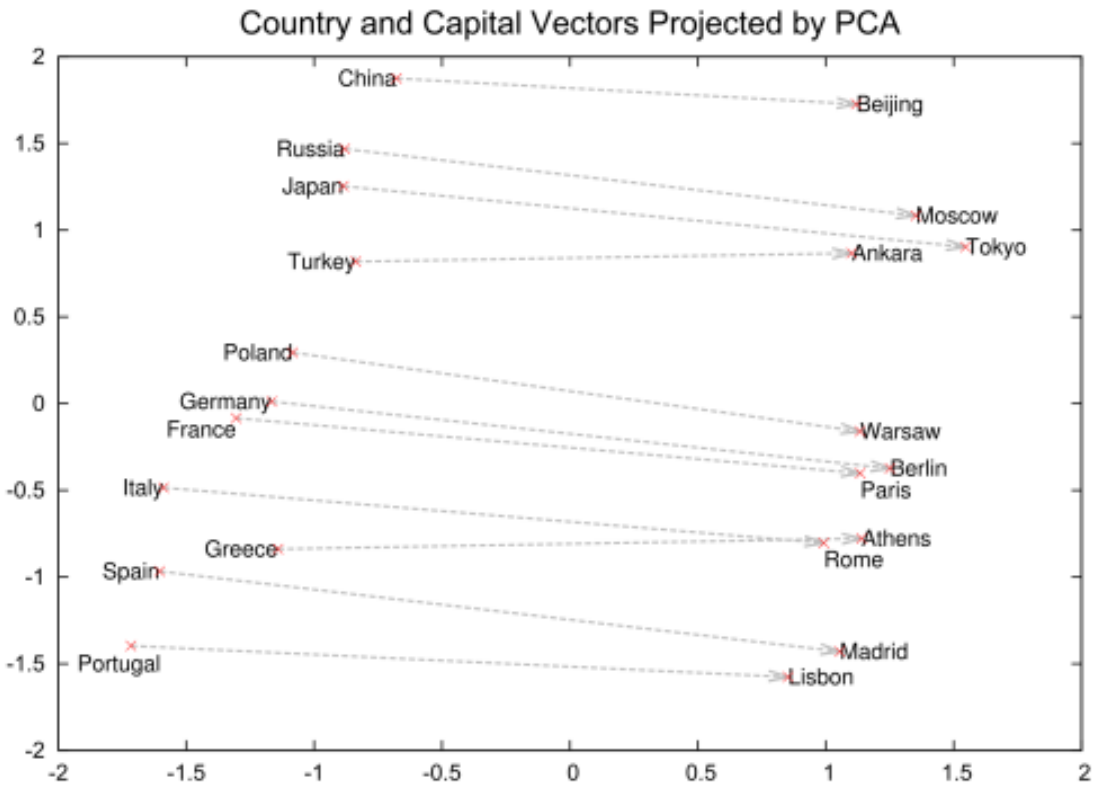


Figure 2: Two-dimensional PCA projection of the 1000-dimensional Skip-gram vectors of countries and their capital cities. The figure illustrates ability of the model to automatically organize concepts and learn implicitly the relationships between them, as during the training we did not provide any supervised information about what a capital city means.

YHK

The Power of Word2Vecs

- They provide a fresh perspective to ALL problems in NLP, and not just solve one problem.
- Technological Improvement
- Rise of deep learning since 2006 (Big Data + GPUs + Work done by Andrew Ng, Yoshua Bengio, Yann Lecun and Geoff Hinton)
- Application of Deep Learning to NLP - led by Yoshua Bengio, Christopher Manning, Richard Socher, Tomas Mikalov
- The need for unsupervised learning . (Supervised learning tends to be excessively dependent on hand-labeled data and often does not scale)

YHK

NLP Activities: How to process text?

Document/Section splitting: Splitting a text into sections

Correlation between three-dimensional ultrasound features and pathological prognostic factors in breast cancer

Jun Jiang · Yaqing Chen · Yichuan Xu · Ming-li Chen · Yinkai Zhu · Wen-bin Guan · Xiaojin Wang

Abstract

Objectives To investigate the correlation of three-dimensional (3D) ultrasound features with prognostic factors in invasive ductal carcinoma.

Methods Surgical resection specimens of 85 invasive ductal carcinomas of 85 women who had undergone 3D ultrasound were included. Morphology features and vascularization pattern on 3D ultrasound were evaluated. Pathologic prognostic factors, including tumour size, histological grade, lymph node status, oestrogen and progesterone receptor status (ER, PR), c-erbB-2 and p53 expression, and microvessel density (MVD) were determined. Correlations of 3D ultrasound features and prognostic factors were analysed.

Results The retraction pattern in the coronal plane had a significant value as an independent predictor of a small tumour size ($P<0.014$), a lower histological grade ($P=0.009$) and positive ER or PR expression status ($P=0.001$, 0.044). The retraction pattern with a hyperechoic ring only existed in low-grade and ER-positive tumours. The presence of the hyperechoic ring strengthened the ability of the retraction pattern to predict a good prognosis of breast cancer. The increased intratumour vascularization index (VI, the mean tumour vascularity) reflected a higher histological grade ($P=0.023$) and had a positive correlation with MVD ($r=0.130$, $P=0.008$).

Conclusions The retraction pattern and histogram indices of VI provided by 3D ultrasound may be useful in predicting prognostic information about breast cancer.

Key Points

- Three-dimensional ultrasound can potentially provide prognostic evaluation of breast cancer.
- The retraction pattern and hyperechoic ring in the coronal plane suggest good prognosis.
- The increased intratumour vascularization index reflects a higher histological grade.
- The intratumour vascularization index is positively correlated with microvessel density.

Keywords Breast · Neoplasms · Ultrasound · Three-dimensional · Prognostic factors

Introduction

The three strongest prognostic factors in invasive breast cancer are widely accepted to be the size of tumour, histological grade and lymph node stage. The larger tumour size (>2 cm), high nuclear grade, and lymph node-positive status usually predict the aggressive biological behaviour with a high recurrence rate and a low survival rate. In addition, the tumour size and lymph node status greatly influence the choice of operative procedure and the decision to administer neoadjuvant chemotherapy [1, 2].

Biological markers such as oestrogen receptors (ER), progesterone receptors (PR), human epidermal growth factor receptor 2 (c-erbB-2) and the p53 index can also be used for prediction of medical treatment response and patient prognosis. The presence of ER and PR in breast cancer always

determines the application of antihormonal therapy and usually indicates a good prognosis. Expression of c-erbB-2 or the p53 index is a powerful and independent prognostic factor for lymph node metastasis and tumour infiltration [1, 3]. Microvessel density (MVD) is the current reference standard in the characterization of tumour angiogenesis and has been shown to be associated with tumour growth, invasion, metastasis and disease-specific survival [4].

Three-dimensional (3D) ultrasound can afford additional information such as morphology features on the coronal plane and a global appearance of the mass vascularity, which cannot be achieved with conventional ultrasound. Therefore, it has been increasingly considered as an important imaging modality for evaluating primary breast cancer. However, so far, 3D ultrasound has been used mainly to differentiate benign and malignant lesions; no reports address correlations between the 3D ultrasound features and prognostic factors [5–7]. We therefore investigated possible correlations between the 3D ultrasound characteristics of invasive ductal carcinoma with pathologic prognostic factors to determine whether 3D ultrasound could be useful in the non-invasive prognostic evaluation of breast cancer.

Materials and methods

Patients

This retrospective study was approved by the ethical standards of the institutional ethics committee, and informed consent was obtained from all patients.

From September 2011 to May 2013, 85 patients with 85 lesions, pathologically proven to be invasive ductal carcinoma, were included in this study. The exclusion criteria were pregnancy or lactation, administration of preoperative chemotherapies or adjuvant chemotherapies. Patients with a breast mass larger than 5 cm were also excluded because more than one 3D volume acquisition was necessary to include the whole lesion plus 3 mm surrounding the breast lesion. All patients were female and aged 26 to 90 years (mean age, 56.3 years).

Ultrasound examination

All ultrasound images were obtained with one type of system (GE Voluson E8 Expert, Zipf, Austria) by two radiologists with 7–12 years of experience in breast ultrasound. An 8–12 LD linear transducer with a frequency of 5–12 MHz was used for 2D ultrasound, and an RSP6-16-D dedicated volume transducer with a frequency of 6–12 MHz was used for 3D ultrasound.

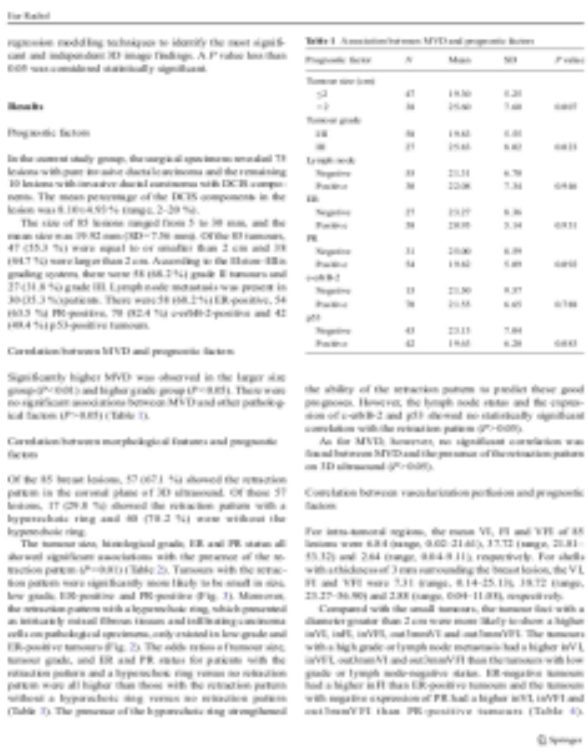
Ultrasound examination was performed with patients in the supine position with elevated arms. Once the breast lesion was detected and the region of interest had been identified, the volume box was superimposed and set to include the entire display section so as to cover the lesion and maximum amount of normal surrounding tissue. The sweep angle was adjusted to 15–29° according to the size of the breast lesion. Then the ultrasound probe was held still with its orthogonality to contact the skin gently. The volume mode was switched on and the 3D ultrasound volume was generated by the automatic rotation of the mechanical transducer. When the first ultrasound examination was finished, the power Doppler mode was added for the second examination and the fixed pretrofital power Doppler settings used were: 0.1 kHz pulse repetition frequency, "low 1" wall motion filter, −2.6 gain and high frequency. The first examination for 3D grayscale imaging took 10–20 s and the second, for 3D power Doppler imaging, took 25–45 s, depending on the size of the tumour. Then the total acquisition time for 3D ultrasound was about 1–2 min. The entire examination was saved in DICOM format and stored on the hard disk for further analysis.

Image analysis

The 3D ultrasound images were reviewed for this analysis by another two radiologists with 6–10 years of experience in breast ultrasound and characterized by consensus. In addition, the radiologists had not performed the data acquisition and were blinded to the patients' clinical and examinographic findings.

The ultrasound image was opened by using the 4D View software. Firstly, the tomographic ultrasound imaging (TUI) was used for a slice by slice documentation in the coronal plane. Then, the volume contrast imaging (VCI) and the surface render mode were added for better observation of the lesion and the surrounding tissue. All the slices were carefully observed to identify the presence of the retraction pattern in the surrounding tissue and the margin of the lesion. The retraction pattern was defined as the hyperechoic straight lines that radiated perpendicularly from the surface of the solid nodule, producing a stellar pattern [8, 9] (Fig. 1). The presence of the retraction pattern was further divided into with or without a hyperechoic ring, which was displayed as an echogenic halo ring between the mass and the surrounding tissue in the coronal plane (Fig. 2a).

The 3D power Doppler imaging analyses were performed using a virtual organ computer-aided analysis (VOCAL) imaging program (GE, Zipf, Austria), which could automatically calculate the histogram indices of vascularization index (VI), flow index (FI) and vascularization flow index (VFI). VI represents the vessels in the defined volume by measuring the number of colour voxels in the region of interest, i.e. the mean tumour vascularity. FI represents the average intensity of flow by measuring the mean colour value in the colour voxels, i.e. the mean blood flow volume; VFI represents both



regression modelling techniques to identify the most significant and independent 3D image findings. A P value less than 0.05 was considered statistically significant.

Results

Prognostic factors

In the current study group, the surgical specimens revealed 73 lesions with pure invasive ductal carcinoma and the remaining 10 lesions with invasive ductal carcinoma with DCIS components. The mean percentage of the DCIS components in the lesion was 8.16±4.95 % (range, 2–20 %).

The size of 85 lesions ranged from 5 to 30 mm, and the mean size was 19.82 mm (SD = 7.86 mm). Of the 85 tumours, 47 (55.3 %) were equal to or smaller than 2 cm and 38 (44.7 %) were larger than 2 cm. According to the Elston–Ellis grading system, there were 58 (68.2 %) grade II tumours and 27 (31.8 %) grade III. Lymph node metastasis was present in 30 (35.3 %) patients. There were 58 (68.2 %) ER-positive, 54 (63.5 %) PR-positive, 70 (82.4 %) c-erbB-2-positive and 42 (49.4 %) p53-positive tumours.

Correlation between MVD and prognostic factors

Significantly higher MVD was observed in the larger size group ($P<0.01$) and higher grade group ($P<0.05$). There was no significant associations between MVD and other pathological factors ($P>0.05$) (Table 1).

Correlation between morphological features and prognostic factors

Of the 85 breast lesions, 57 (67.1 %) showed the retraction pattern in the coronal plane of 3D ultrasound. Of these 57 lesions, 17 (29.8 %) showed the retraction pattern with a hyperechoic ring and 40 (70.2 %) were without the hyperechoic ring.

The tumour size, histological grade, ER and PR status all showed significant associations with the presence of the retraction pattern ($P=0.01$) (Table 2). Tumours with the retraction pattern were significantly more likely to be small in size, low grade, ER-positive and PR-positive (Fig. 3). Moreover, the retraction pattern with a hyperechoic ring, which presented as intricately mixed fibrous tissues and infiltrating carcinoma cells on pathological specimens, only existed in low-grade and ER-positive tumours (Fig. 2). The odds ratios of tumour size, tumour grade, and ER and PR status for patients with the retraction pattern and a hyperechoic ring versus no retraction pattern were all higher than those with the retraction pattern without a hyperechoic ring versus no retraction pattern (Table 3). The presence of the hyperechoic ring strengthened

Table 1 Association between MVD and prognostic factors

Prognostic factor	N	Mean	SD	P value
Tumour size (cm)				
≤2	47	19.30	8.21	
>2	38	25.60	7.68	0.007
Tumour grade				
II	58	19.82	8.01	
III	27	25.63	8.02	0.023
Lymph node				
Negative	55	21.31	8.70	
Positive	30	22.06	7.31	0.940
ER				
Negative	27	23.27	8.36	
Positive	58	20.95	8.34	0.931
PR				
Negative	31	23.00	8.39	
Positive	54	19.82	8.09	0.058
c-erbB-2				
Negative	15	23.80	9.37	
Positive	70	21.55	8.05	0.748
p53				
Negative	43	23.13	7.89	
Positive	42	19.65	8.20	0.043

the ability of the retraction pattern to predict these good prognoses. However, the lymph node status and the expression of c-erbB-2 and p53 showed no statistically significant correlation with the retraction pattern ($P>0.05$).

As for MVD, however, no significant correlation was found between MVD and the presence of the retraction pattern on 3D ultrasound ($P>0.05$).

Correlation between vascularization perfusion and prognostic factors

For intratumoral regions, the mean VI, FI and VFI of 85 lesions were 6.84 (range, 0.02–21.61), 37.72 (range, 21.81–53.32) and 2.64 (range, 0.04–9.11), respectively. For shells with a thickness of 3 mm surrounding the breast lesion, the VI, FI and VFI were 7.31 (range, 0.14–25.13), 38.72 (range, 23.27–56.90) and 2.88 (range, 0.04–11.03), respectively.

Compared with the small tumours, the tumour foci with a diameter greater than 2 cm were more likely to show a higher inVI, inFI, inVFI, outSmmVI and outSmmVFI. The tumours with a high grade or lymph node metastasis had a higher inVI, inVFI, outSmmVI and outSmmVFI than the tumours with low grade or lymph node-negative status. ER-negative tumours had a higher inFI than ER-positive tumours and the tumours with negative expression of PR had a higher inVI, inVFI and outSmmVFI than PR-positive tumours (Table 4).

Sentence splitting

Sentence splitting: Splitting a text into sentences

11 Sentences (= "T-" or "Terminable" units *only* if independent clauses are puctuated as separate sentences, e.g. "I came and he went"-->"I came. And he went.")
Average 23.55 words (SD=12.10)

OBJECTIVES: To investigate the correlation of three-dimensional (3D) ultrasound features with prognostic factors in invasive ductal carcinoma.

METHODS: Surgical resection specimens of 85 invasive ductal carcinomas of 85 women who had undergone 3D ultrasound were included.

Morphology features and vascularization perfusion on 3D ultrasound were evaluated.

Pathologic prognostic factors, including tumour size, histological grade, lymph node status, oestrogen and progesterone receptor status (ER, PR), c erbB-2 and p53 expression, and microvessel density (MVD) were determined.

Correlations of 3D ultrasound features and prognostic factors were analysed.

RESULTS: The retraction pattern in the coronal plane had a significant value as an independent predictor of a small tumour size (P #8201;= 0.014), a lower histological grade (P #8201;= 0.009) and positive ER or PR expression status (P #8201;= 0.001, 0.044).

The retraction pattern with a hyperechoic ring only existed in low-grade and ER-positive tumours.

The presence of the hyperechoic ring strengthened the ability of the retraction pattern to predict a good prognosis of breast cancer.

The increased intra-tumour vascularization index (VI, the mean tumour vascularity) reflected a higher histological grade (P #8201;= 0.025) and had a positive correlation with MVD (r #8201;= 0.530, P #8201;= 0.001).

CONCLUSIONS: The retraction pattern and histogram indices of VI provided by 3D ultrasound may be useful in predicting prognostic information about breast cancer.

KEY POINTS: • Three-dimensional ultrasound can potentially provide prognostic evaluation of breast cancer. • The retraction pattern and hyperechoic ring in the coronal plane suggest good prognosis. • The increased intra-tumour vascularization index reflects a higher histological grade. • The intra-tumour vascularization index is positively correlated with microvessel density.

Built by Lextutor, SD function added 18 Nov 2010

YHK

Tokenization

Tokenization

- Process of breaking a stream of text up into tokens (= words, phrases, symbols, or other meaningful elements)
- Typically performed at the "word" level
- Not easy: Hewlett-Packard, U.S.A., in some languages there is no "space" between words!

YHK

Stemming

Stemming

- Reduces similar words to a given "stem"
- E.g. detects, detected, detecting, detect : detect (stem).
- Usually set of rules for suffix stripping
- Most popular for English: Porter's Algorithm
- 36% reduction in indexing vocabulary (English)
- Linguistic correctness of resulting stems not necessary (sensitivities : sensit)

YHK

Lemmatization

Lemmatization

- Uses a vocabulary and full morphological analysis of words
- Aims to remove inflectional endings only
- Return the base or dictionary form of a word, which is known as the lemma.
- E.g. saw : see,
  been, was : be

YHK

Part-of-speech tagging

Part-of-speech tagging: Assigning a syntatic tag to each word in a sentence

Stanford Parser

Please enter a sentence to be parsed:

Surgical resection specimens of 85 invasive ductal carcinomas of 85 women who had undergone 3D ultrasound were included.

Language: English · Sample Sentence Parse

Your query

Surgical resection specimens of 85 invasive ductal carcinomas of 85 women who had undergone 3D ultrasound were included.

Tagging

```
Surgical/NNP resection/NN specimens/NNS of/IN 85/CD invasive/JJ
ductal/JJ carcinomas/NNS of/IN 85/CD women/NNS who/WP had/VBD
undergone/VBN 3D/CD ultrasound/NN were/VBD included/VBN ./.
```

http://nlp.stanford.edu:8080/corenlp/

YHK

Parsing

Parsing: Building the syntactic tree of a sentence

Parse

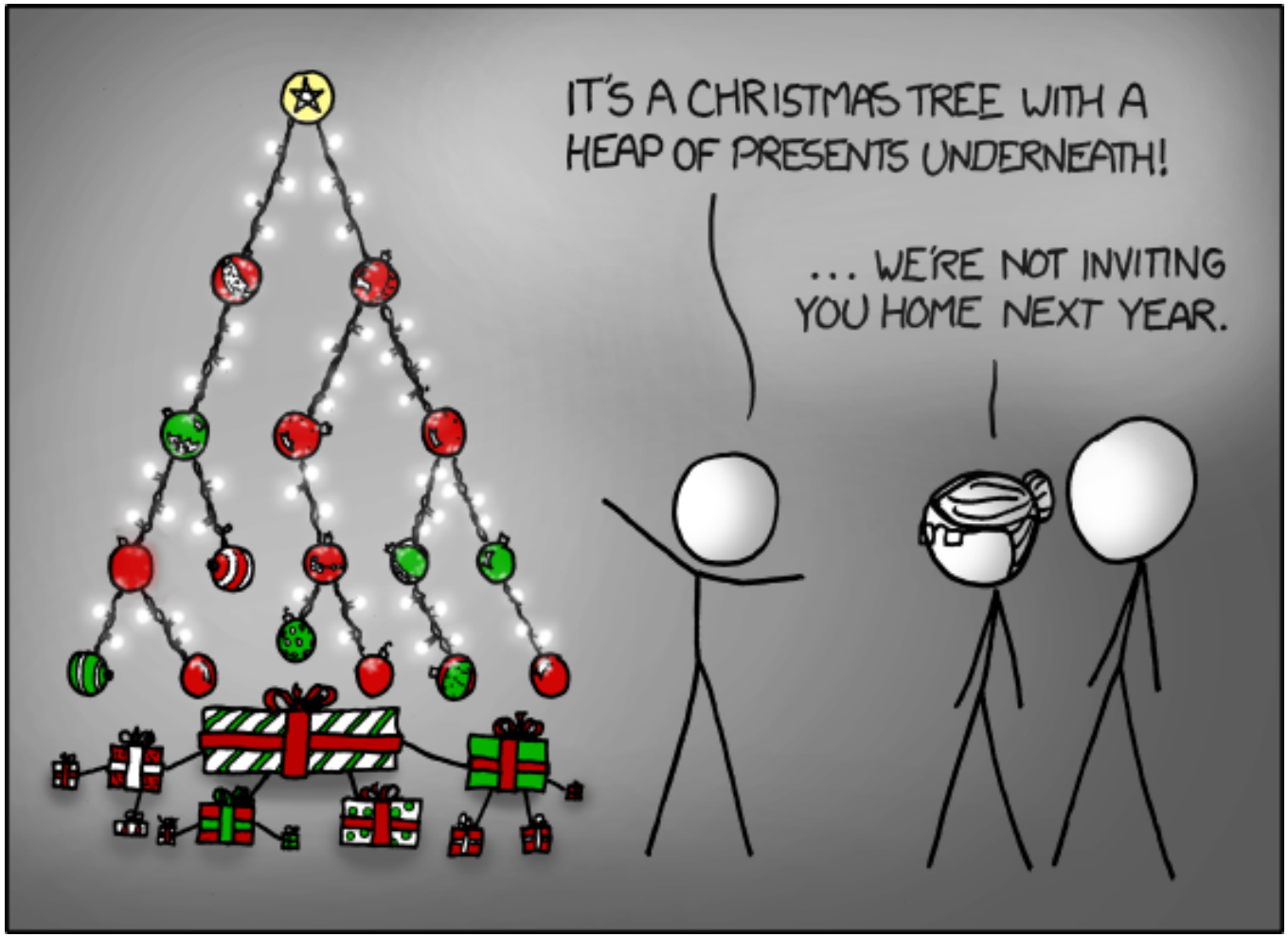
```
(ROOT
  (S
    (NP
      (NP (NNP Surgical) (NN resection) (NNS specimens))
      (PP (IN of)
        (NP
          (NP (CD 85) (JJ invasive) (JJ ductal) (NNS carcinomas))
          (PP (IN of)
            (NP
              (NP (CD 85) (NNS women))
              (SBAR
                (WHNP (WP who))
                (S
                  (VP (VBD had)
                    (VP (VBN undergone)
                      (NP (CD 3D) (NN ultrasound))))))))))
    (VP (VBD were)
      (VP (VBN included)))
    (. .)))
```

http://nlp.stanford.edu:8080/corenlp/

YHK

Parsing

$((DaimlerChryslersshares)_{NP}(rose(threeeights)_{NUMP}(to22)_{PP-NUM})_{VP})_{S}$



YHK

Syntax Tree

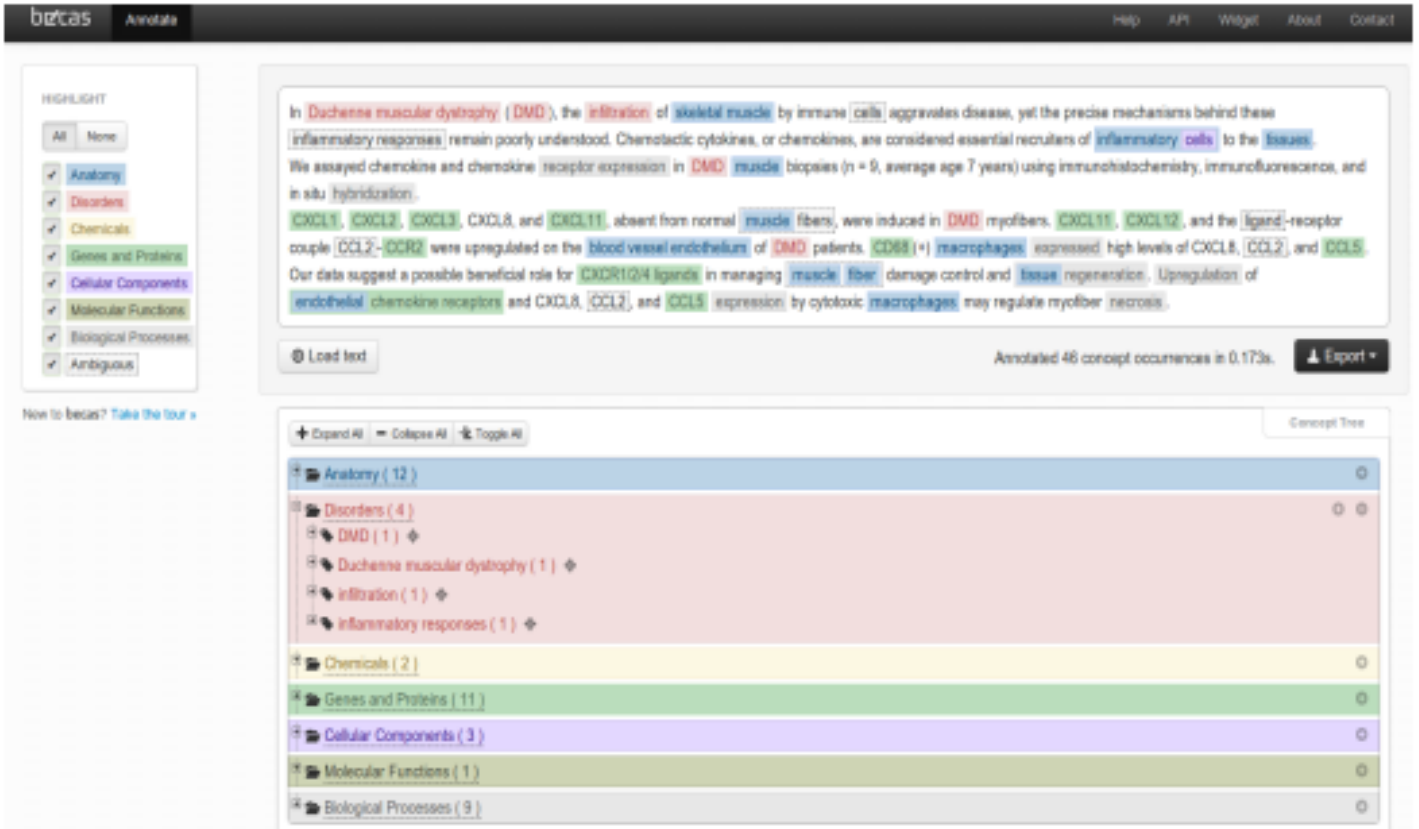
Syntax: Sample English grammar

$S \rightarrow NP\ VP$	$Det \rightarrow that \mid this \mid a$
$S \rightarrow Aux\ NP\ VP$	$Noun \rightarrow book \mid flight \mid meal \mid money$
$S \rightarrow VP$	$Verb \rightarrow book \mid include \mid prefer$
$NP \rightarrow Pronoun$	$Pronoun \rightarrow I \mid she \mid me$
$NP \rightarrow Proper\text{-}Noun$	$Proper\text{-}Noun \rightarrow Houston \mid TWA$
$NP \rightarrow Det\ Nominal$	$Aux \rightarrow does$
$Nominal \rightarrow Noun$	$Preposition \rightarrow from \mid to \mid on \mid near \mid through$
$Nominal \rightarrow Nominal\ Noun$	
$Nominal \rightarrow Nominal\ PP$	
$VP \rightarrow Verb$	
$VP \rightarrow Verb\ NP$	
$VP \rightarrow Verb\ NP\ PP$	
$VP \rightarrow Verb\ PP$	
$VP \rightarrow VP\ PP$	
$PP \rightarrow Preposition\ NP$	

YHK

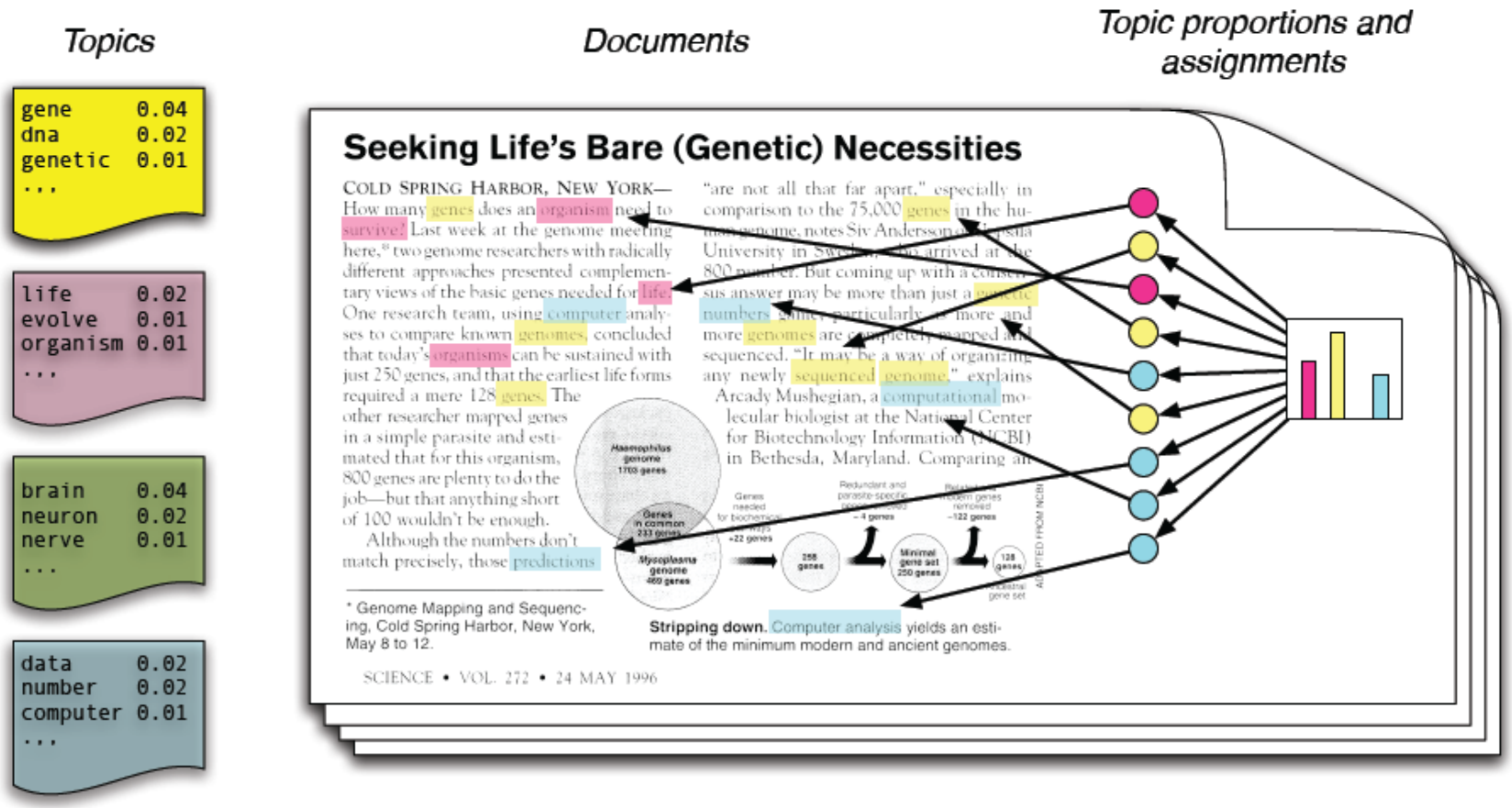
Named-entity recognition

Named-entity recognition: Identifying pre-defined entity types in a sentence



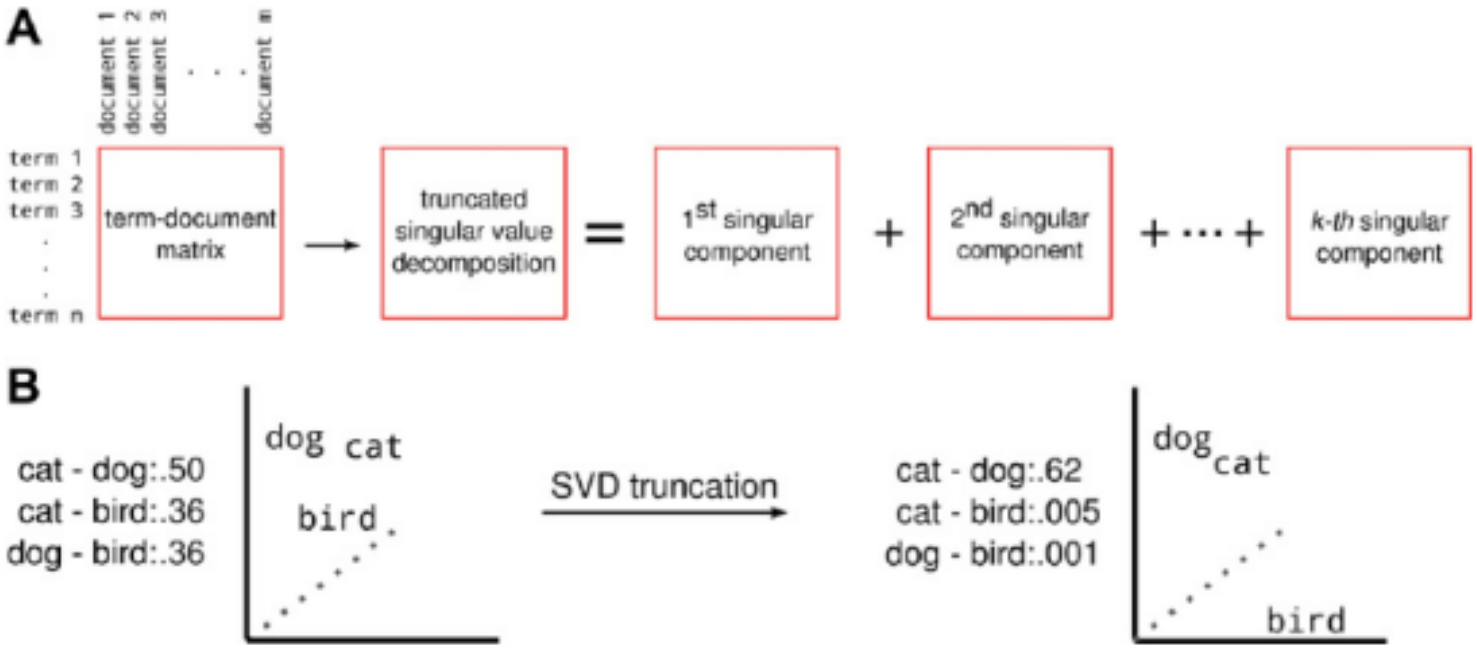
Topic modelings

Topic modeling: Identifying structures in the text corpus



Word embeddings

Word embeddings: Compute a vector representing the distributed representation for every word



NLP with Spacy

YHK

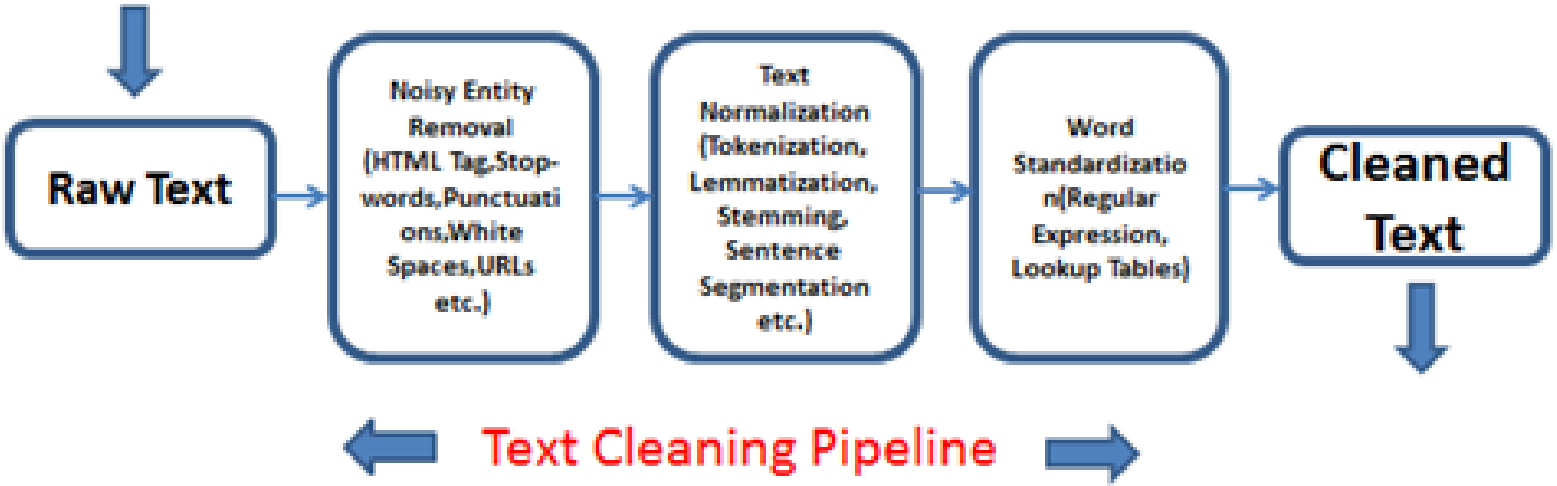
Introduction

- Natural Language Processing (NLP) and Machine Learning (ML) form the foundation for modern intelligent systems.
- Retrieval Augmented Generation (RAG) leverages NLP pipelines, embeddings, and ML for knowledge-grounded responses.
- This presentation explores:
  - Text preprocessing and representation
  - spaCy-based NLP components
  - ML techniques for classification and clustering
  - Embedding-based search and semantic understanding
  - Sentiment analysis applications

(Ref: Stanford NLP Group)

Text Preprocessing

- Clean and normalize raw text data for consistent downstream analysis.
- Common steps include lowercasing, punctuation & digit removal, and whitespace normalization.
- Ensures reliable tokenization and model training.
- Helps retain semantic integrity by focusing on meaningful linguistic content.



(Ref: Text Cleaning Steps ResearchGate)

YHK

Tokenization

- Tokenization splits text into smaller linguistic units (words, subwords, or sentences).
- spaCy provides efficient rule-based and statistical tokenizers.
- Basis for vectorization, tagging, and parsing.

```
import spacy
nlp = spacy.load("en_core_web_sm")
doc = nlp("Retrieval-Augmented Generation is powerful!")
tokens = [token.text for token in doc]
print(tokens)
# ['Retrieval', '-', 'Augmented', 'Generation', 'is', 'powerful', '!']
```

YHK

Lowercasing & Cleaning

- Lowercasing removes case sensitivity in NLP models.
- Cleaning eliminates noise: URLs, HTML tags, punctuation, numbers.
- Helps search and RAG systems achieve higher recall and consistency.

```
import re
text = "Visit https://openai.com for <b>AI Research!</b>"
cleaned = re.sub(r"http\S+|<.*?>", "", text).lower()
print(cleaned)
# 'visit  for ai research!'

```

YHK

Stop Word Removal

- Removes high-frequency words that add little meaning.
- Reduces dimensionality and improves efficiency of embeddings.
- Example stop words: “the”, “is”, “in”, “on”.

```
from spacy.lang.en.stop_words import STOP_WORDS
tokens = ["Retrieval", "Augmented", "Generation", "is", "powerful"]
filtered = [w for w in tokens if w.lower() not in STOP_WORDS]
print(filtered)
# ['Retrieval', 'Augmented', 'Generation', 'powerful']
```

YHK

Stemming & Lemmatization

- Stemming removes suffixes using heuristic rules.
- Lemmatization uses grammar-based transformations to return dictionary form.
- Reduces vocabulary size and enhances text matching in retrieval.

```
from nltk.stem import WordNetLemmatizer
lem = WordNetLemmatizer()
print(lem.lemmatize("running", "v"))  # 'run'
print(lem.lemmatize("better", "a"))   # 'good'

```

YHK

Regular Expressions

- Enable pattern-based extraction from text.
- Examples: extract emails, phone numbers, or URLs.
- Essential for data cleaning, anonymization, and text mining.

```
import re
text = "Contact us at info@company.com or sales@domain.org"
emails = re.findall(r"\b[A-Za-z0-9._%+-]+@[A-Za-z]+\.[A-Z|a-z]{2,}\b", text)
print(emails)
# ['info@company.com', 'sales@domain.org']

```

YHK

Project: Text Preprocessing Pipeline

- Modular preprocessing pipeline for flexible text workflows.
- Each function handles one cleaning step.
- Easily extendable for lemmatization, stemming, or entity masking.

```
class TextPipeline:
    def __init__(self, steps):
        self.steps = steps  # list of functions

    def run(self, text):
        for fn in self.steps:
            text = fn(text)
        return text

pipeline = TextPipeline([basic_clean])
print(pipeline.run("Hello World! This is 2025."))
```

YHK

Project: Document Cleaner

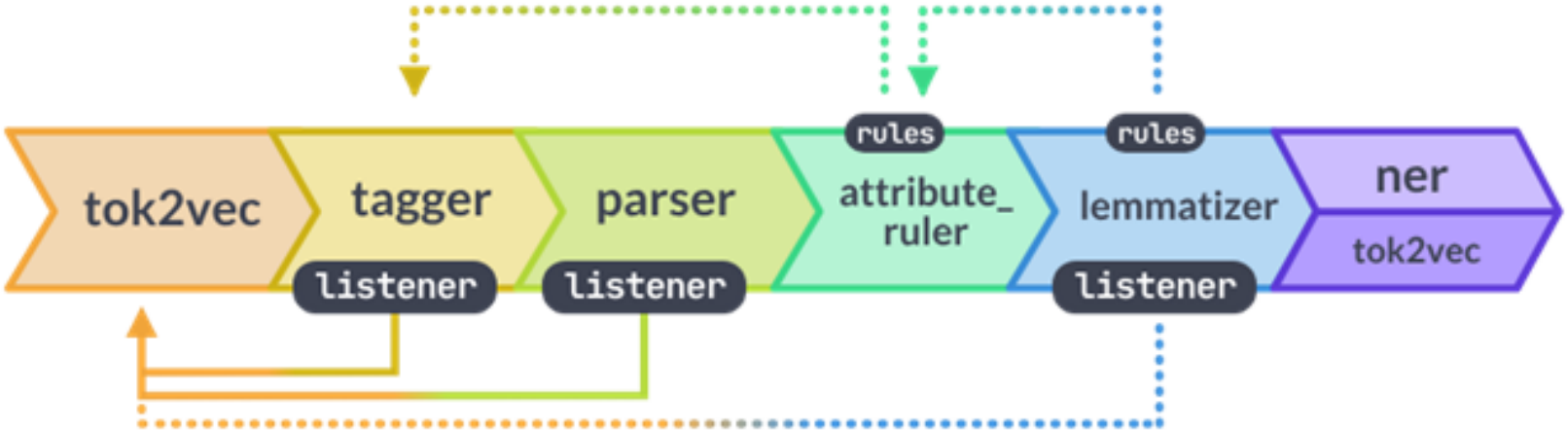
- Extracts text from PDFs, HTML, and Word documents.
- Normalizes whitespace, removes headers/footers, handles encoding.
- Outputs clean text ready for embedding or indexing.

```
from bs4 import BeautifulSoup
from PyPDF2 import PdfReader

def clean_document(path):
    if path.endswith(".pdf"):
        text = "".join([page.extract_text() for page in PdfReader(path).pages])
    elif path.endswith(".html"):
        html = open(path).read()
        text = BeautifulSoup(html, "html.parser").get_text()
    else:
        text = open(path).read()
    return re.sub(r"\s+", " ", text.strip())

```

YHK

spaCy Pipeline Basics

- spaCy's pipeline includes tokenization, tagging, parsing, and entity recognition.
- Designed for efficiency and extensibility.
- Supports custom components for preprocessing and postprocessing.



(Ref: spaCy Docs)

Part-of-Speech Tagging

- Assigns syntactic roles: noun, verb, adjective, etc.
- Enables grammar-aware retrieval, information extraction, and text summarization.

```
doc = nlp("NLP models enhance communication.")
for token in doc:
    print(token.text, token.pos_)
# Output: NLP NOUN, models NOUN, enhance VERB, communication NOUN
```

YHK

Named Entity Recognition (NER)

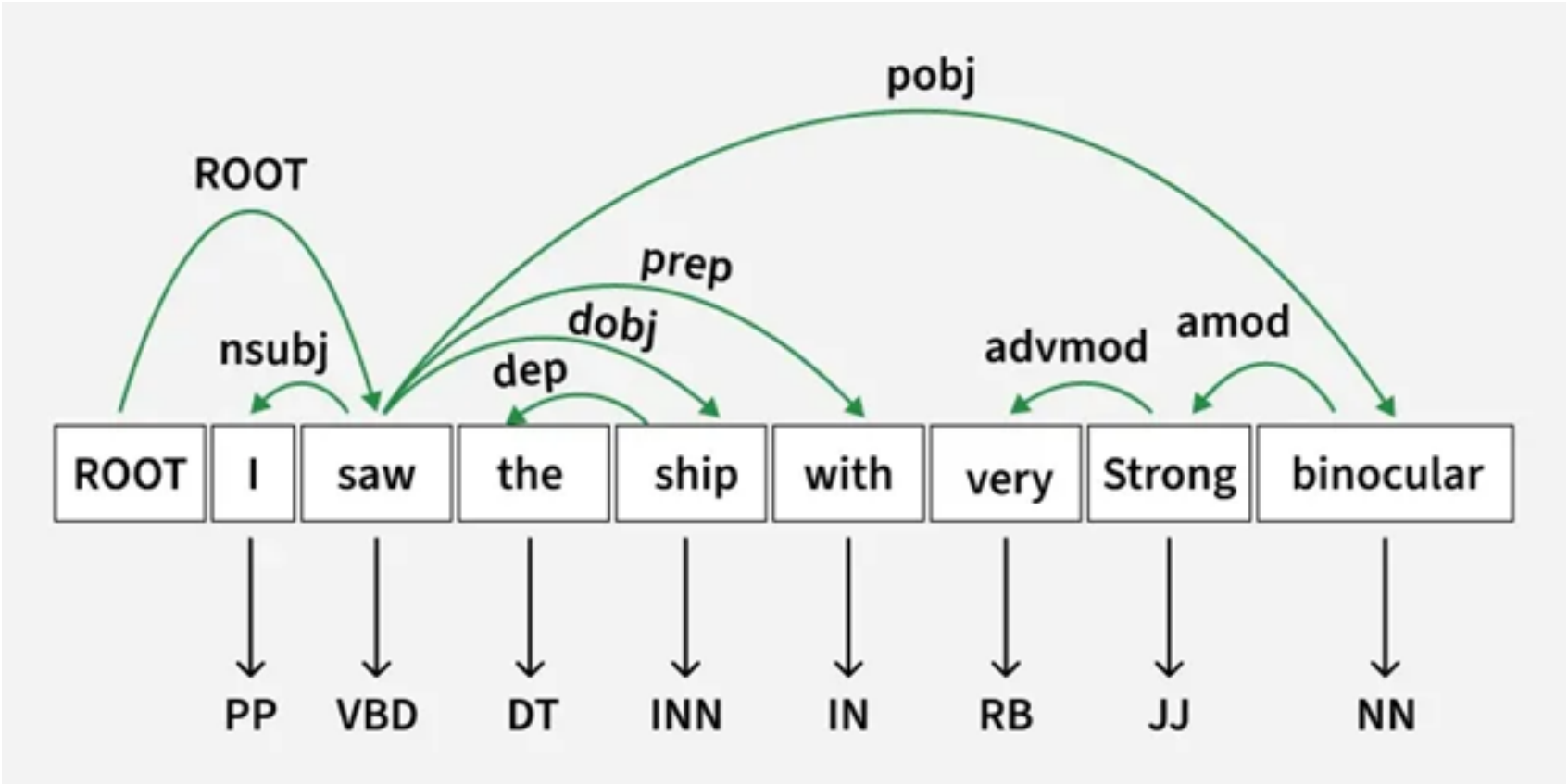
- Identifies named entities: PERSON, ORG, LOC, DATE, MONEY, etc.
- Crucial for knowledge extraction and linking to external databases.

```
doc = nlp("OpenAI was founded in San Francisco in 2015.")
for ent in doc.ents:
    print(ent.text, ent.label_)
# OpenAI ORG, San Francisco GPE, 2015 DATE
```

YHK

Dependency Parsing

- Analyzes grammatical structure of sentences.
- Reveals subject-object relationships.
- Basis for relation extraction and semantic role labeling.



(Ref: Dependency Parsing with NLTK - GeeksforGeeks)

YHK

Text Similarity

- Measures semantic closeness between two texts.
- Useful in RAG for ranking documents by contextual relevance.

```
doc1 = nlp("AI improves healthcare")
doc2 = nlp("Artificial intelligence helps medicine")
print(doc1.similarity(doc2))
# Output: similarity score ~0.85
```

YHK

Project: Named Entity Extractor

- Extracts and exports named entities from text.
- Can be integrated with visualization tools or stored in databases.

```
def extract_entities(text):
    nlp = spacy.load("en_core_web_sm")
    doc = nlp(text)
    return {ent.text: ent.label_ for ent in doc.ents}

print(extract_entities("Elon Musk leads SpaceX and Tesla."))
```

YHK

Project: Text Similarity Engine

- Uses embeddings to compute cosine similarity between texts.
- Enables semantic document search and clustering.

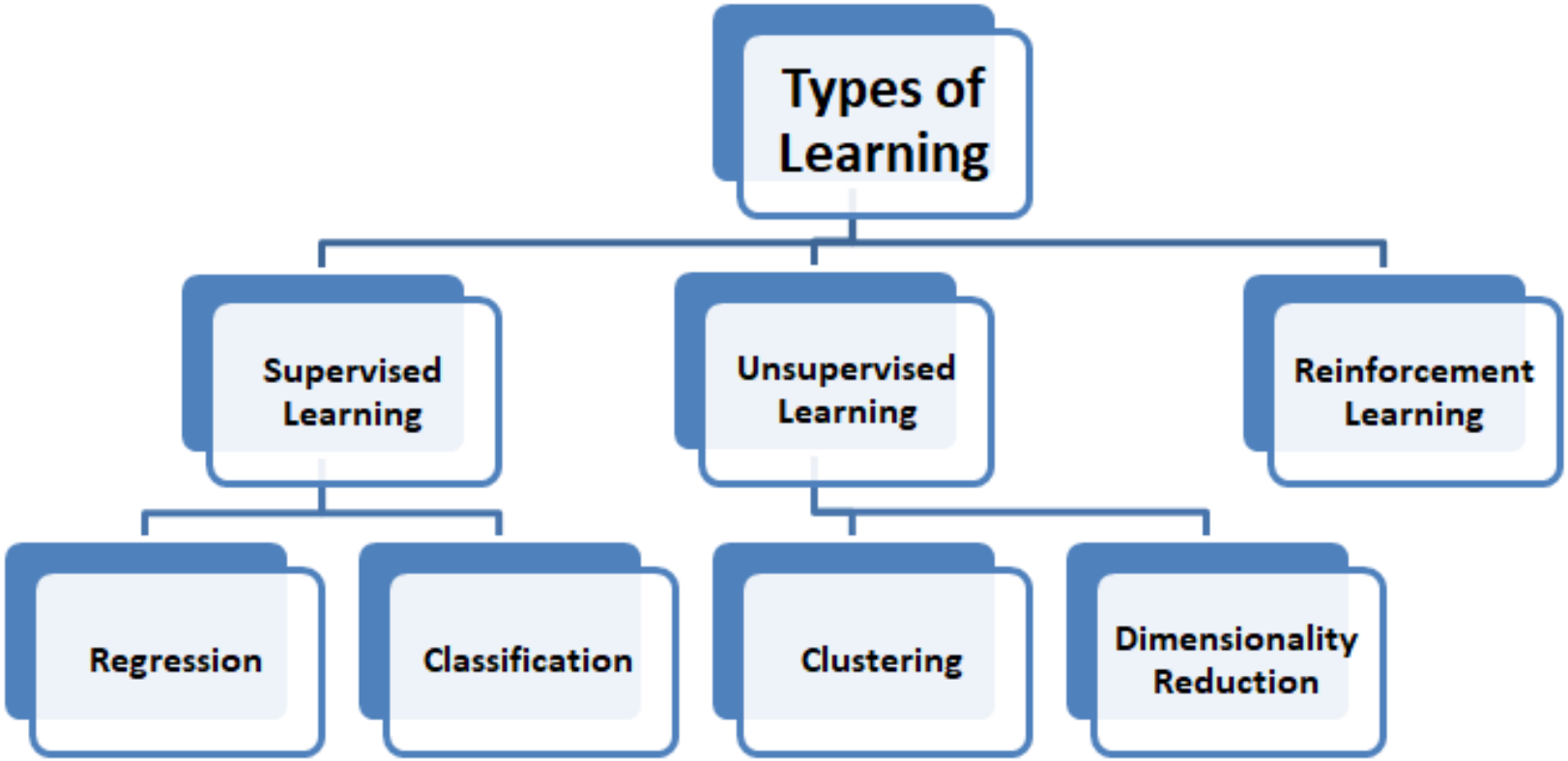
```
from sklearn.metrics.pairwise import cosine_similarity

def rank_docs(query, docs, model):
    q_vec = model.encode([query])
    d_vecs = model.encode(docs)
    scores = cosine_similarity(q_vec, d_vecs)[0]
    ranked = sorted(zip(docs, scores), key=lambda x: x[1], reverse=True)
    return ranked[:3]
```

YHK

Machine Learning

Types of Machine Learning

- **Supervised**: learns from labeled data (e.g., text classification).
- **Unsupervised**: discovers hidden structure (e.g., clustering).
- **Reinforcement**: learns from feedback/reward.
- NLP tasks map naturally to these paradigms.



(Ref: https://www.studytrigger.com/article/types-of-learning-in-machine-learning/)

YHK

Classification

- Predicts categorical labels (spam, positive/negative sentiment).
- Logistic Regression is a strong baseline for text classification.

```
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.linear_model import LogisticRegression

vec = TfidfVectorizer()
X_train = vec.fit_transform(train_texts)
model = LogisticRegression(max_iter=200)
model.fit(X_train, y_train)
preds = model.predict(vec.transform(test_texts))
```

Clustering

- Groups similar documents based on content.
- Common methods: KMeans, DBSCAN, Agglomerative.
- Supports topic discovery and unsupervised organization.

```
from sklearn.cluster import KMeans
from sklearn.decomposition import PCA
import matplotlib.pyplot as plt

km = KMeans(n_clusters=5, random_state=42)
km.fit(embeddings)
labels = km.labels_

pca = PCA(n_components=2)
reduced = pca.fit_transform(embeddings)
plt.scatter(reduced[:,0], reduced[:,1], c=labels)
plt.show()
```

YHK

Project: Iris Flower Classifier

- Classic ML dataset demonstration.
- Illustrates feature scaling, model training, and evaluation.

```
from sklearn.datasets import load_iris
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score

X, y = load_iris(return_X_y=True)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3)
model = LogisticRegression(max_iter=300)
model.fit(X_train, y_train)
print("Accuracy:", accuracy_score(y_test, model.predict(X_test)))
```

YHK

Project: Document Clusterer

- Embeds documents using spaCy or sentence-transformers.
- Applies KMeans for unsupervised grouping.
- Visualizes using t-SNE or PCA for semantic clusters.

```
import spacy
from sklearn.cluster import KMeans
nlp = spacy.load("en_core_web_md")

docs = ["AI in healthcare", "Quantum computing basics", "AI in finance"]
vectors = [nlp(d).vector for d in docs]
labels = KMeans(n_clusters=2).fit_predict(vectors)
print(list(zip(docs, labels)))
```

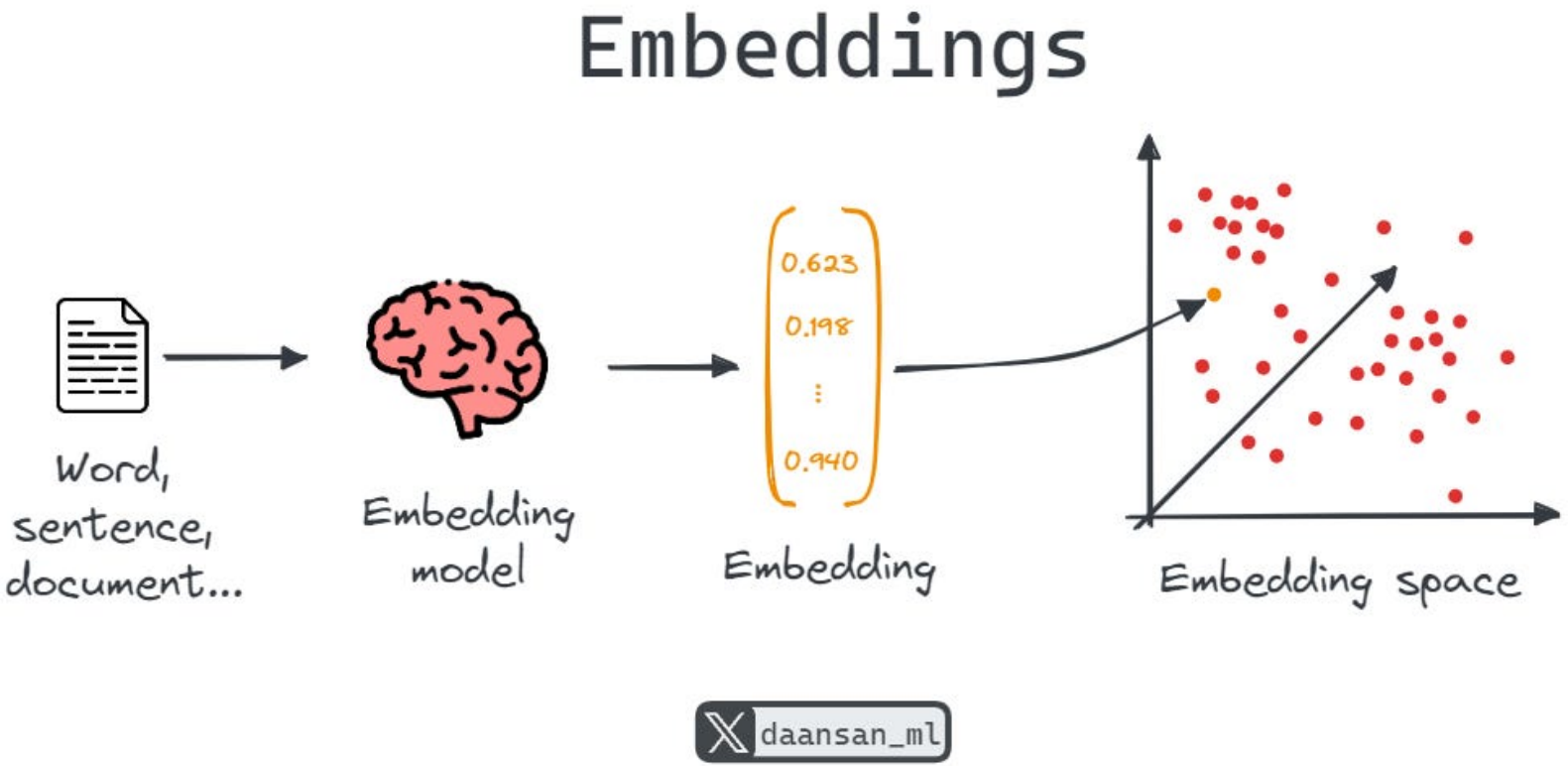
YHK

Word Embeddings

YHK

Word Embeddings Concepts

- Represent words as dense numerical vectors capturing context.
- Models: Word2Vec, GloVe, FastText, Transformer embeddings.
- Enable semantic understanding and transfer learning.



(Ref: https://mlpills.substack.com/p/issue-58-embeddings-in-nlp)

Project: Semantic Search Engine

- Retrieve documents semantically, not just by keyword match.
- Uses embedding models and cosine similarity for ranking.

```
from sentence_transformers import SentenceTransformer, util
model = SentenceTransformer("all-MiniLM-L6-v2")

def semantic_search(query, docs):
    q_emb = model.encode(query, convert_to_tensor=True)
    d_emb = model.encode(docs, convert_to_tensor=True)
    scores = util.cos_sim(q_emb, d_emb)
    return sorted(zip(docs, scores[0]), key=lambda x: float(x[1]),
     reverse=True)[:3]
```

YHK

Project: Word Analogy Solver

- Demonstrates vector arithmetic: $king - man + woman \approx queen$
- Captures latent gender, tense, and semantic relations.

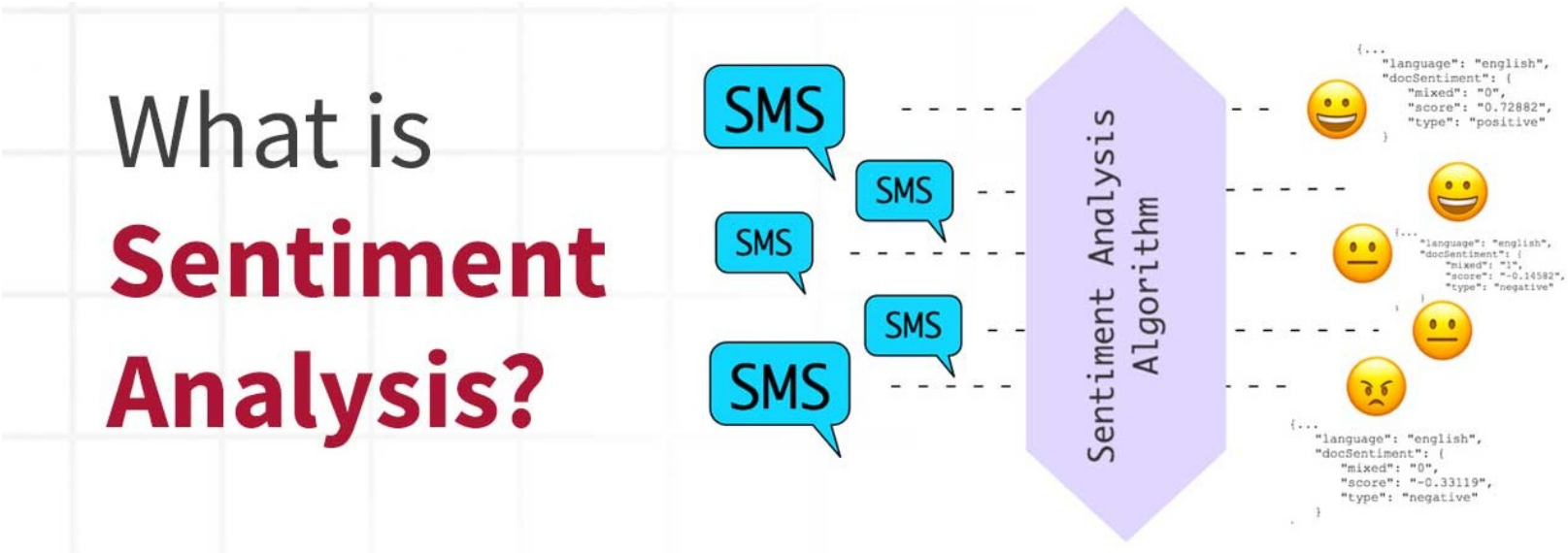
```
from gensim.models import KeyedVectors
model = KeyedVectors.load_word2vec_format("GoogleNews-vectors.bin",
    binary=True)
result = model.most_similar(positive=["king", "woman"], negative=["man"],
    topn=1)
print(result)  # [('queen', 0.75)]
```

YHK

Sentiment Analysis

Sentiment Analysis Basics

- Detects emotional polarity (positive, neutral, negative).
- Used in reviews, feedback, and social media monitoring.
- Can be rule-based, ML-based, or transformer-based.



(Ref: Sentiment Analysis — Engati)

YHK

Project: Review Sentiment Analyzer

- Train a sentiment classifier using Naive Bayes.
- Vectorize text with TF-IDF.
- Evaluate accuracy using test set.

```
from sklearn.naive_bayes import MultinomialNB
from sklearn.metrics import accuracy_score

X = vec.fit_transform(reviews)
model = MultinomialNB()
model.fit(X, labels)
preds = model.predict(vec.transform(test_reviews))
print("Accuracy:", accuracy_score(test_labels, preds))
```

YHK

Project: Tweet Sentiment Dashboard

- Streams tweets and classifies their sentiment in real-time.
- Visualizes sentiment trends dynamically using dashboards.

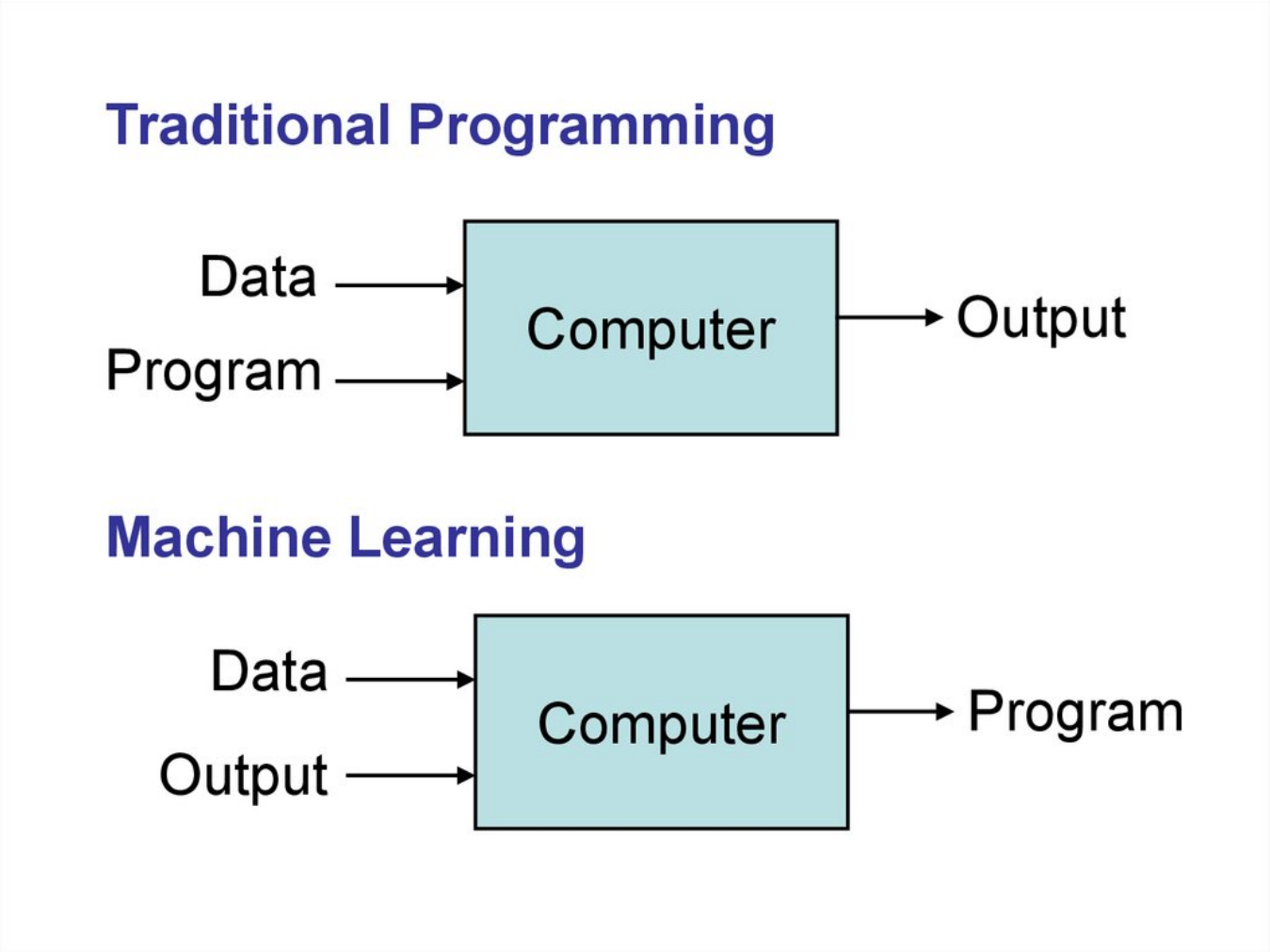
```
import tweepy, pandas as pd
from textblob import TextBlob

def classify_sentiment(text):
    polarity = TextBlob(text).sentiment.polarity
    return "positive" if polarity > 0 else "negative" if polarity < 0 else
      "neutral"

# TODO: integrate with dashboard for visualization
```

YHK

LLMs

Background

Traditional vs. Machine Learning?



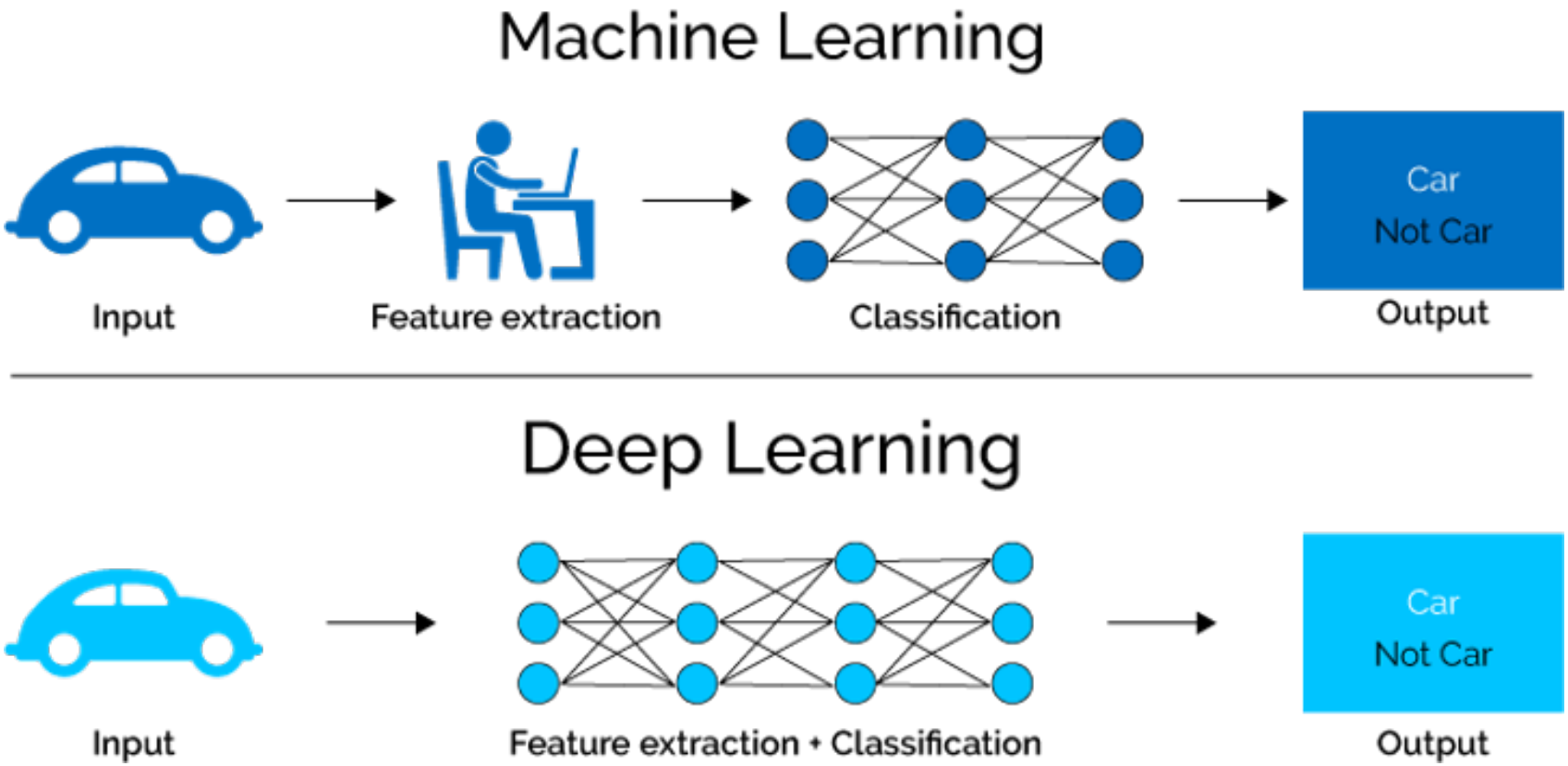
Why Machine Learning?

- Problems with High Dimensionality
- Hard/Expensive to program manually
- Job $$$

YHK

ML vs DL: What's the difference?

Deep learning algorithms attempt to learn (multiple levels of) representation by using a hierarchy of multiple layers



(Reference: https://www.xenonstack.com/blog/static/public/uploads/media/machine-learning-vs-deep-learning.png)

YHK

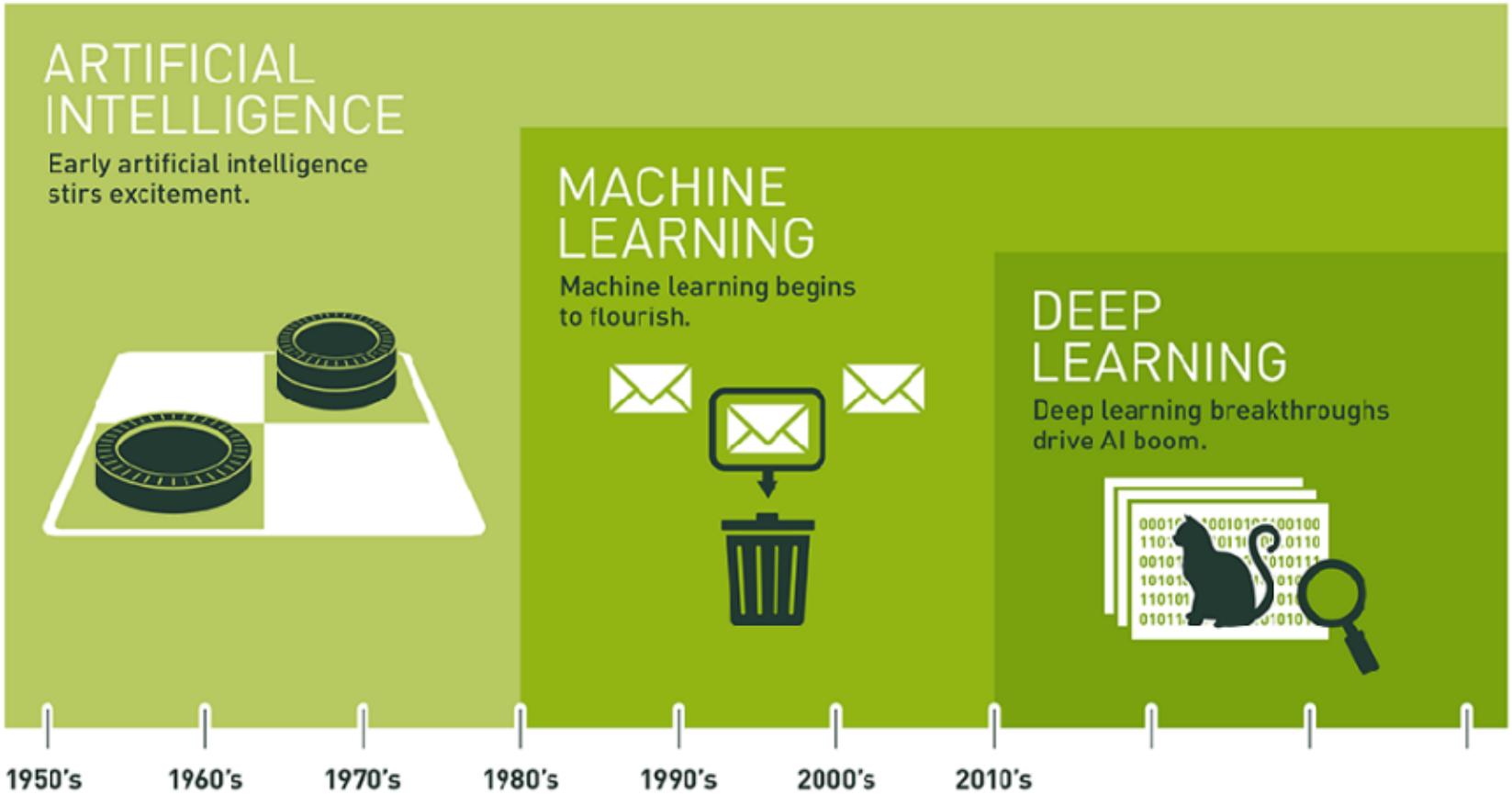
Use Deep Learning When ...

- You have lots of data (about 10k+ examples)
- The problem is "complex" - speech, vision, natural language
- The data is unstructured
- Techniques to model 'ANY' function given 'ENOUGH' data.

(Ref: Introduction to TensorFlow 2.0 - Brad Miro)

YHK

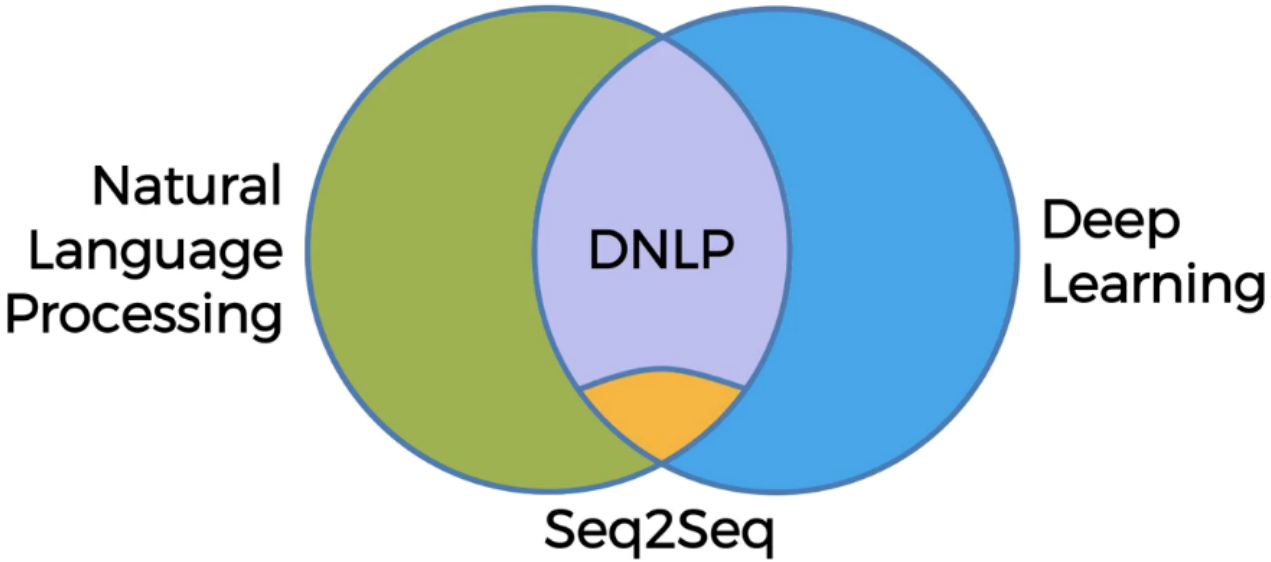
Relationship between AI, ML, DL



(Ref: https://blogs.nvidia.com/blog/2016/07/29/whats-difference-artificial-intelligence-machine-learning-deep-learning-ai/)

YHK

What is Deep NLP



(Ref: Deep Learning and NLP A-Z - Kirill Eremenko)
(Note: Size is not indicative of importance)

Seq2Seq is heavily used technique of DNLP for sequence to sequence modeling, eg Translation, Q & A, etc. Thats the basis of Large Language Models (LLMs)

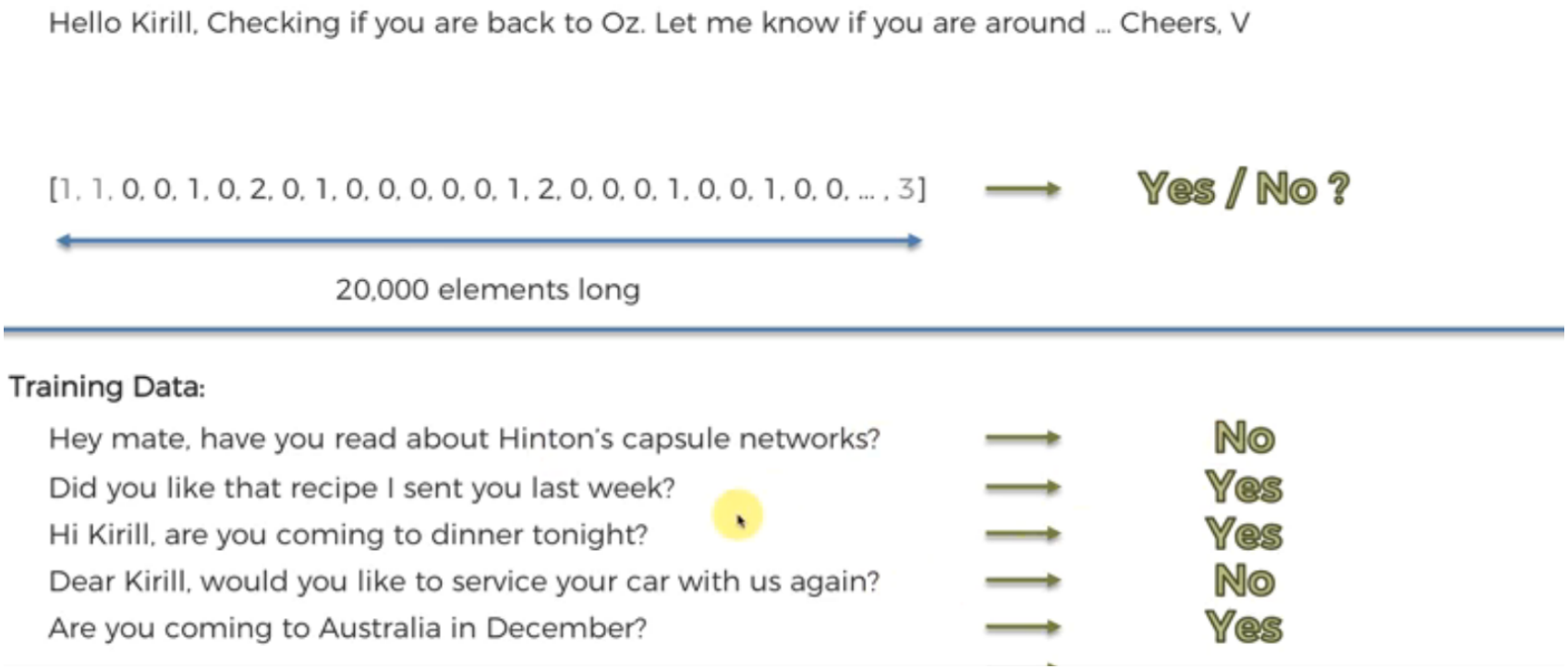
YHK

Overview of Large Language Models

YHK

Typical Machine Learning Classification

- Each text item thus gets converted to fixed size vector, thus features.
- In training, weights are computed based on the given target.
- Once model is ready, it is able to answer target, say, Yes or No to unseen text.



(Ref: Deep Learning and NLP A-Z - Kirill Eremenko)

YHK

Evolution of Vectorization

Vectors can be statistical (frequency based) or Machine/Deep Learning (supervised) based. Simple to complex.

(Ref: Analytics Vidhya https://editor.analyticsvidhya.com/uploads/59483evolution_of_NLP.png)

YHK

How to Vectorize? Representing words by their context

- Distributional semantics: **A word's meaning is given by the words that frequently appear close-by**
  - *"You shall know a word by the company it keeps"* (J. R. Firth 1957)
  - One of the most successful ideas of modern statistical NLP!
- When a word *w* appears in a text, its **context** is the set of words that appear nearby (within a fixed-size window).
- Use the many contexts of *w* to build up a representation of *w*

…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***

(Ref: CS224n: Natural Language Processing with Deep Learning - Christopher Manning)

YHK

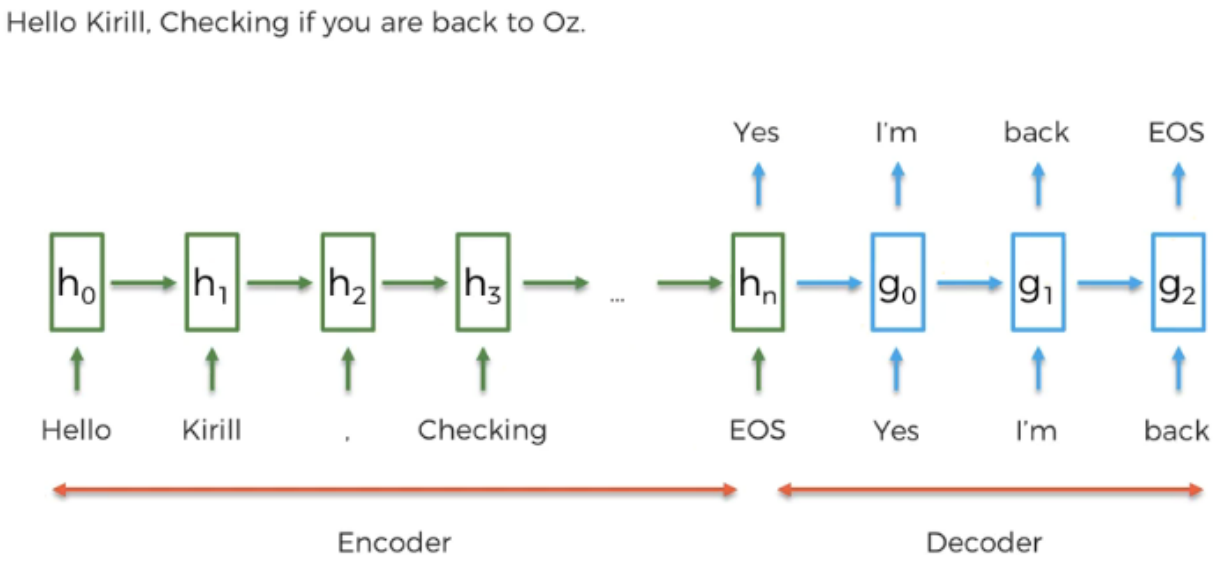
Word vectors

- Dense vector for each word
- Called distributed representation, word embeddings or word representations
- Test: similar to vectors of words that appear in similar contexts

$$\textit{banking} = \begin{pmatrix} 0.286 \\ 0.792 \\ -0.177 \\ -0.107 \\ 0.109 \\ -0.542 \\ 0.349 \\ 0.271 \end{pmatrix}$$

YHK

Seq2Seq architecture



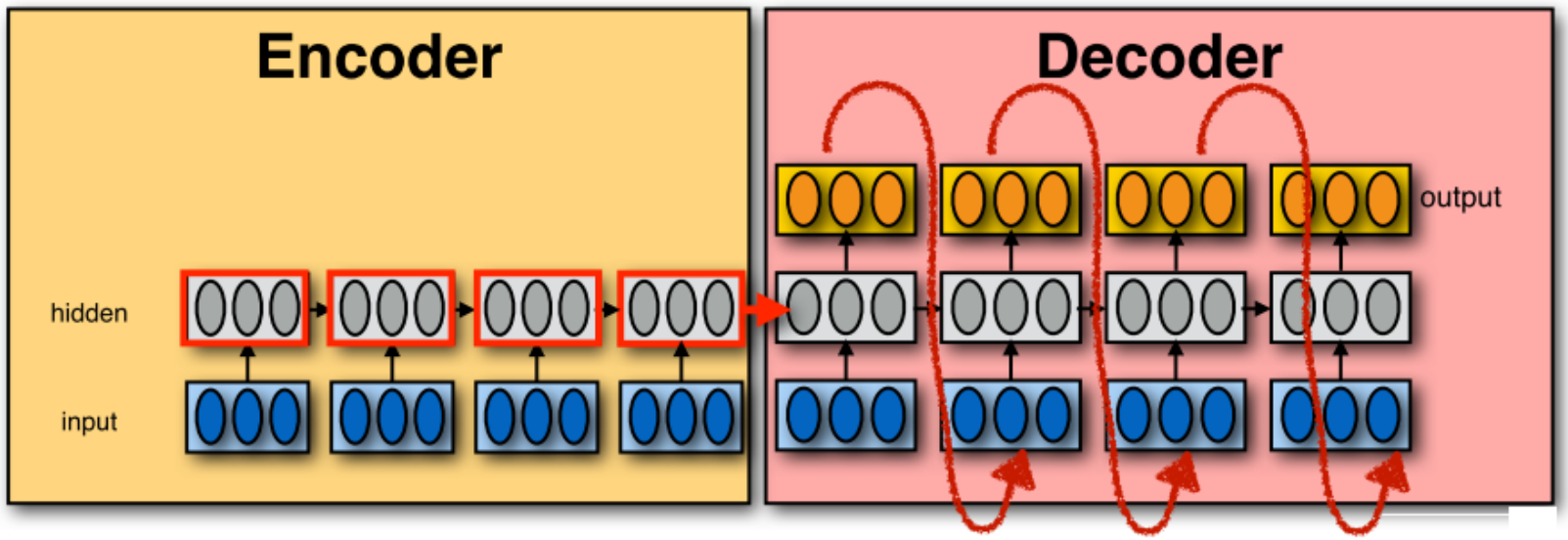
(Ref: Deep Learning and NLP A-Z - Kirill Eremenko)

During training, Encoder is fed with Questions and decoder with Answers. Weights in gates, hidden states get settled. During testing for each sequence of input, encoder results in to a combo vector. Decoder takes this and starts spitting out words one by one, probabilistically.

YHK

Encoder-Decoder (seq2seq) model

- The decoder is a language model that generates an output sequence conditioned on the input sequence.
  - Vanilla RNN: condition on the last hidden state
  - Attention: condition on all hidden states



(Ref: CS447 Natural Language Processing (J. Hockenmaier)

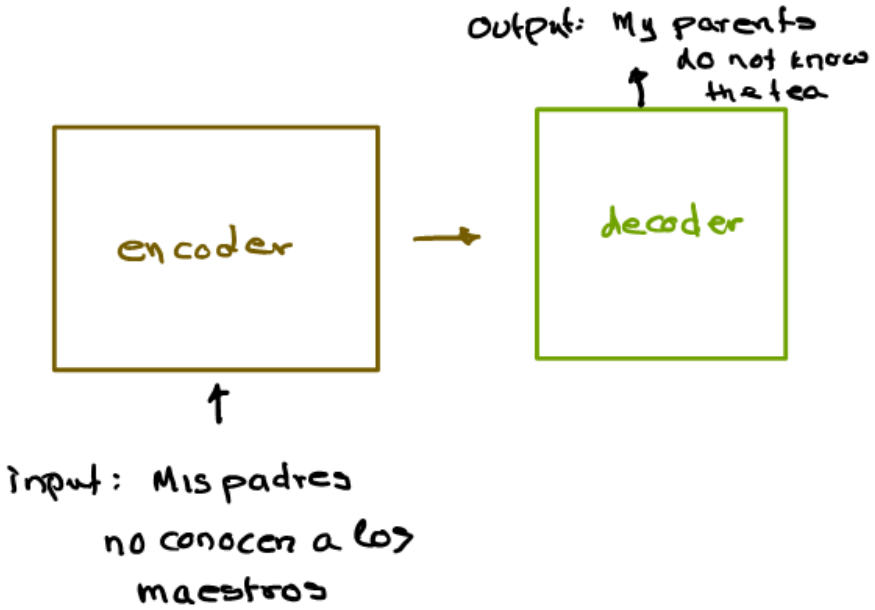
Transformers use Self-Attention

- Attention so far (in seq2seq architectures): In the decoder (which has access to the complete input sequence), compute attention weights over encoder positions that depend on each decoder position
- Self-attention: If the encoder has access to the complete input sequence, we can also compute attention weights over encoder positions that depend on each encoder position

YHK

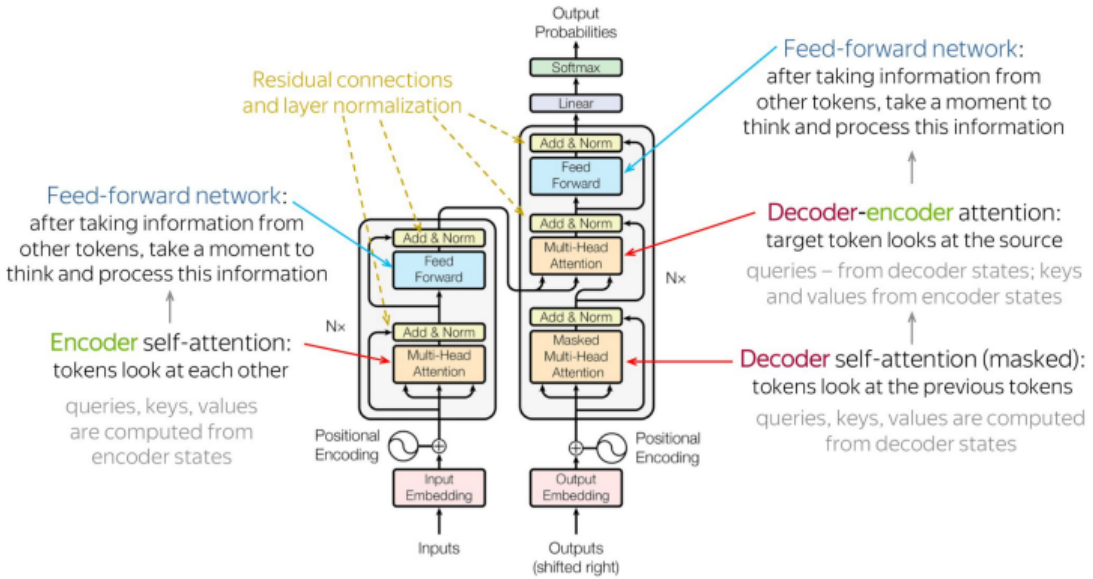
Transformers

- In its heart it contains an encoding component, a decoding component, and connections between them.
- The Transformer is a model that uses attention to boost the speed with which seq2seq with attention models can be trained.
- The biggest benefit, however, comes from how The Transformer lends itself to parallelization. How?



YHK

Transformer Models



(Ref: The Complete Prompt Engineering for AI Bootcamp (2023))

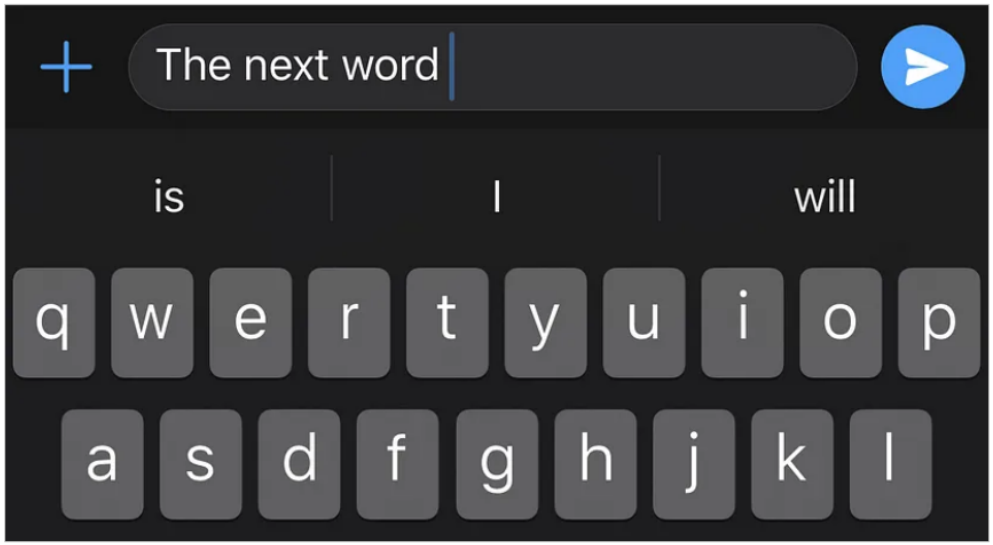
- No recurrence, so parallelization possible
- Context information captured via attention and positional encodings
- Consists of stacks of layers with various sublayers

Transformers are basis of (the most) Large Language Models

YHK

What is a Language Models?

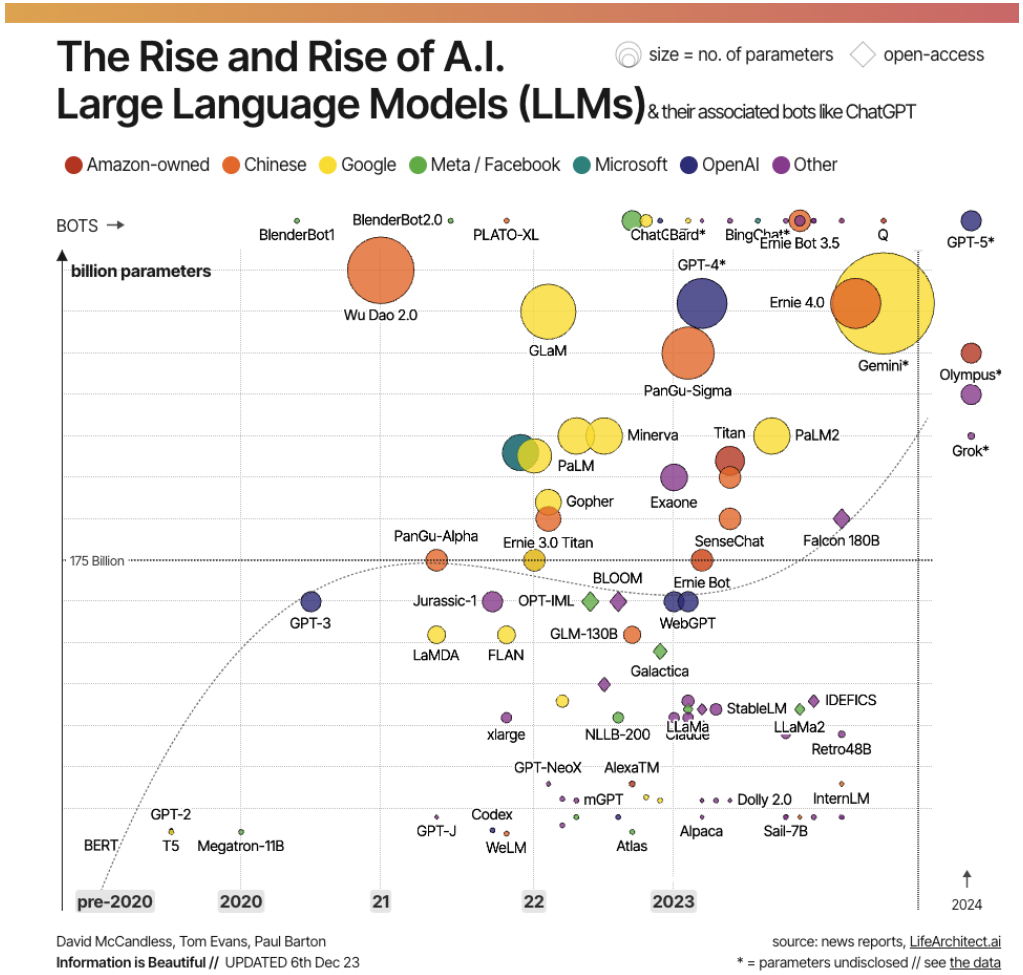
- While typing SMS, have you seen it suggests next word?
- While typing email, have you seen next few words are suggested?
- How does it suggest? (suggestions are not random, right?)
- In the past, for "Lets go for a . . . ', if you have typed 'coffee' 15 times, 'movie' say 4 times, then it learns that. Machine/Statistical Learning.
- Next time, when you type "Lets go for a ", what will be suggested? why?
- This is called Language Model. Predicting the next word. When done continuously, one after other, it spits sentence, called Generative Model.



Next word prediction using language modeling in keyboards(Mandar Deshpande)

YHK

LLM - Information is beautiful



(Ref: https://informationisbeautiful.net/visualizations/the-rise-of-generative-ai-large-language-models-llms-like-chatgpt/)

How LLMs work?

- **Transformer-Based Architecture:**
  - Utilizes the Transformer architecture for processing input sequences.
  - Self-attention mechanism captures long-range dependencies.
- **Pre-training:**
  - Trained on a massive corpus of text data in an unsupervised manner.
  - Learns contextualized representations of words and phrases.
- **Generative Capabilities:**
  - Can generate coherent and contextually relevant text.
  - Useful for a wide range of natural language understanding and generation tasks.
- **Fine-tuning (Optional):**
  - Model can be fine-tuned on specific downstream tasks.
  - Adaptation to user or domain-specific requirements.

YHK

Transformer Architecture

- **Input Representation:** Embedding layer converts input tokens into high-dimensional vectors.
- **Positional Encoding:** Adds positional information to the input embeddings.
- **Multi-Head Self Attention:**
  - Allows each token to focus on different parts of the input sequence.
  - Multiple attention heads capture diverse patterns.
- **Layer Normalization & Residual/Skip Connections:**
  - Stabilizes training using layer normalization.
  - Residual connections help in mitigating vanishing/exploding gradient problems.
- **Encoder-Decoder Structure (for Sequence-to-Sequence tasks):** In tasks like translation, multiple encoder layers process the input, and then multiple decoder layers generate the output.
- **Output Layer:** Produces the final output sequence.

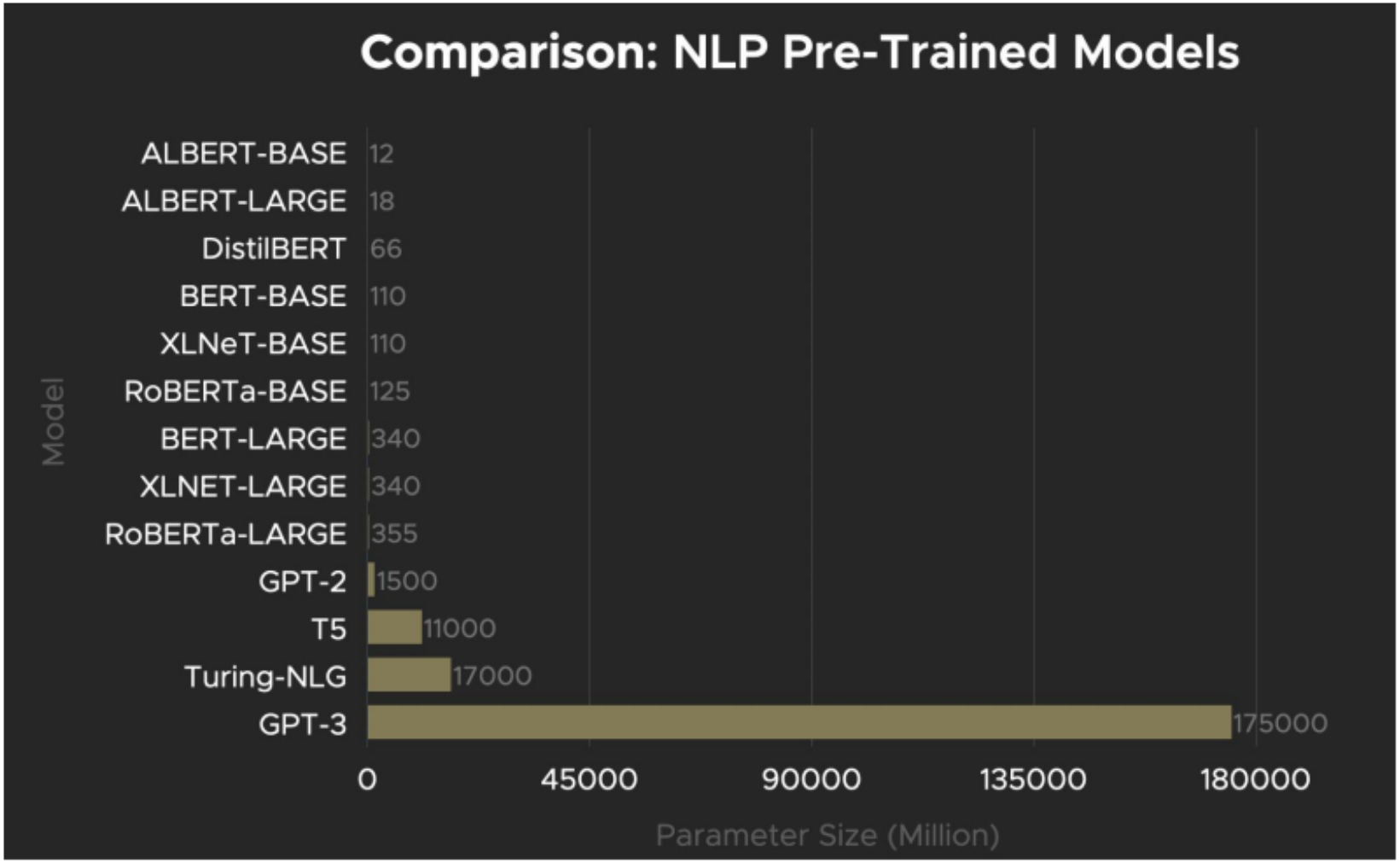
YHK

Decoder-Only Transformers (e.g., GPT)

- **Architecture:**
  - GPT uses a transformer architecture consisting solely of decoder layers.
  - Decoders attend to the entire input sequence during training and generation.
- **Positional Embeddings:** Incorporates position information to handle sequence order.
- **Self-Attention Mechanism with Masking:**
  - Each position attends to all positions in the preceding context.
  - During training, masking ensures that positions after the current one are not considered.
  - Prevents the model from peeking at future tokens during generation.
- **Autoregressive Generation:**
  - Generates output tokens one at a time in an autoregressive manner.
  - Previous tokens influence the generation of subsequent tokens.

YHK

Large Language Models - Comparison



(Ref: Deus.ai https://www.deus.ai/post/gpt-3-what-is-all-the-excitement-about)

YHK

LLM Training

- Training LLMs involves instructing the model to comprehend and generate human-like text.
- **Input Text:** LLMs are exposed to extensive text data from diverse sources like books, articles, and websites.
- During training, the model predicts the next word/token based on context, learning patterns and relationships.
- **Optimizing Weights:** The model has weights for parameters reflecting feature significance.
- Throughout training, weights are fine-tuned to minimize error, enhancing the model's prediction accuracy.
- After initial training, LLMs can be customized for tasks using small sets of supervised data. This process is known as **fine-tuning**.
- **Fine-tuning Parameters:** LLMs adjust parameter values based on error feedback during predictions.
- The model refines its language understanding by iteratively adjusting parameters, improving token prediction accuracy.
- Training may vary for specific LLM types, like those optimized for continuous text or dialogue.

YHK

LLM Performance Factors

- **Model Architecture:** LLM performance is influenced by the design and intricacy of its architecture.
- **Dataset Quality:** The quality and diversity of the training dataset shape the model's language understanding.
- Training a private LLM demands substantial computational resources and expertise.
- Duration ranges from days to weeks, contingent on model complexity and dataset size.
- Cloud-based solutions and high-performance GPUs expedite the training process.
- LLM training is meticulous and resource-intensive, forming the basis for language comprehension and generation.

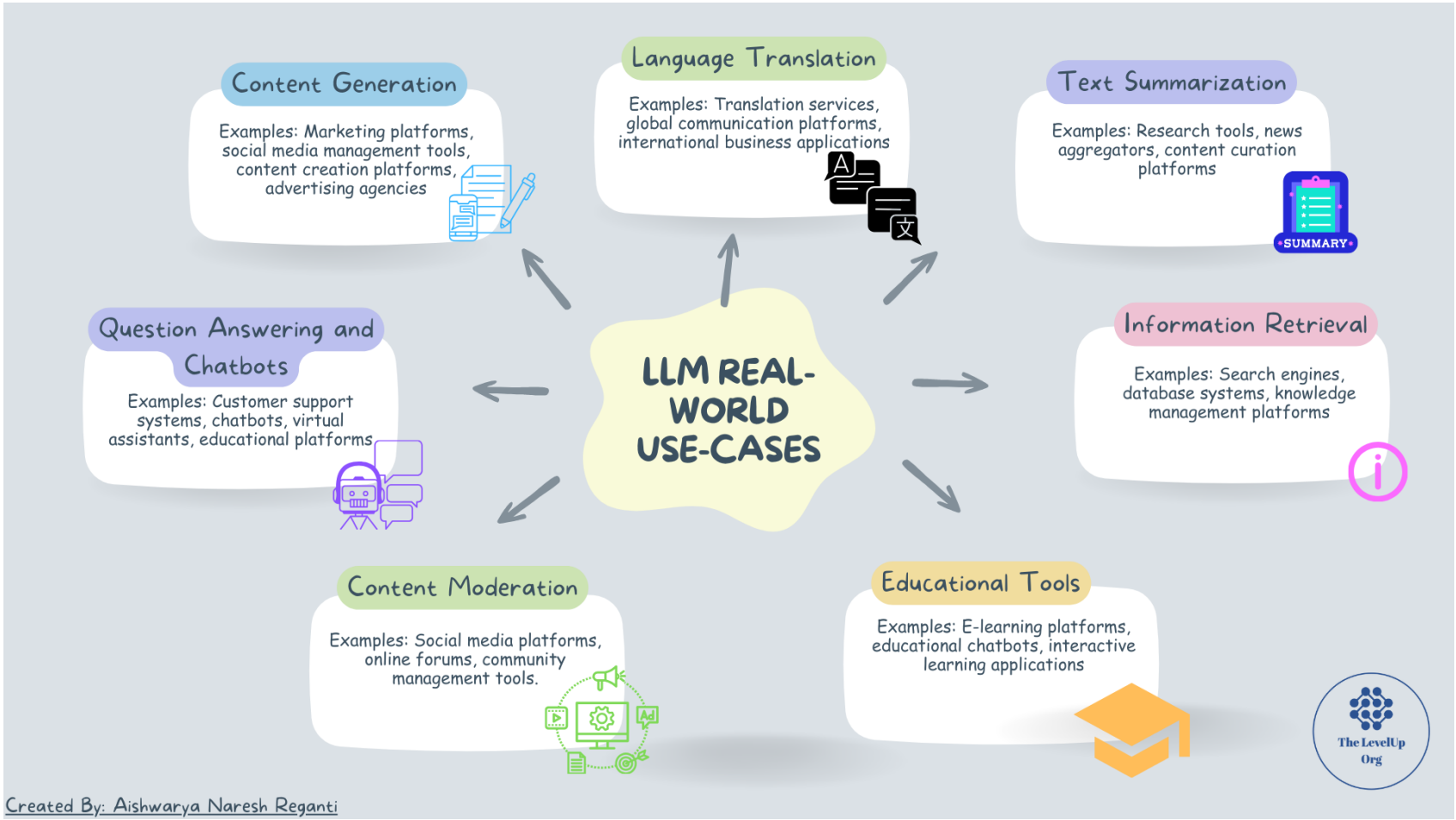
YHK

Decoder-Only Transformers (e.g., GPT)

- **Architecture:**
  - GPT uses a transformer architecture consisting solely of decoder layers.
  - Decoders attend to the entire input sequence during training and generation.
- **Positional Embeddings:** Incorporates position information to handle sequence order.
- **Self-Attention Mechanism with Masking:**
  - Each position attends to all positions in the preceding context.
  - During training, masking ensures that positions after the current one are not considered.
  - Prevents the model from peeking at future tokens during generation.
- **Autoregressive Generation:**
  - Generates output tokens one at a time in an autoregressive manner.
  - Previous tokens influence the generation of subsequent tokens.

YHK

LLM Real World Use Cases



(Ref: Applied LLMs Mastery 2024 - Aishwarya Reganti)

LLM Real World Use Cases

No.	Use case	Description
1	Content Generation	Craft human-like text, videos, code and images when provided with instructions
2	Language Translation	Translate languages from one to another
3	Text Summarization	Summarize lengthy texts, simplifying comprehension by highlighting key points.
4	Question Answering and Chatbots	LLMs can provide relevant answers to queries, leveraging their vast knowledge
5	Content Moderation	Assist in content moderation by identifying and filtering inappropriate or harmful language
6	Information Retrieval	Retrieve relevant information from large datasets or documents.
7	Educational Tools	Tutor, provide explanations, and generate learning materials.

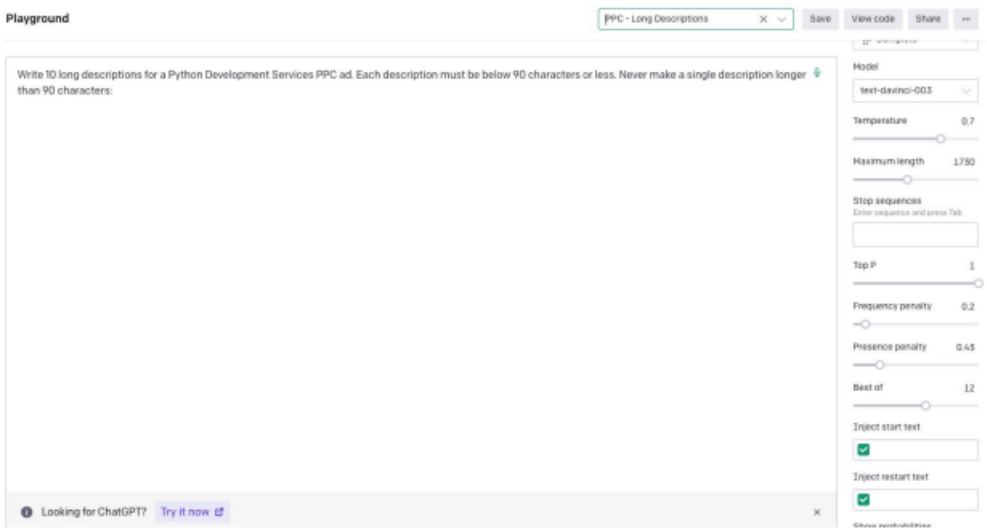
(Ref: Applied LLMs Mastery 2024 - Aishwarya Reganti)

YHK

LLM Challenges

(Ref: Applied LLMs Mastery 2024 - Aishwarya Reganti)

ChatGPT — GPT3.5/GPT4

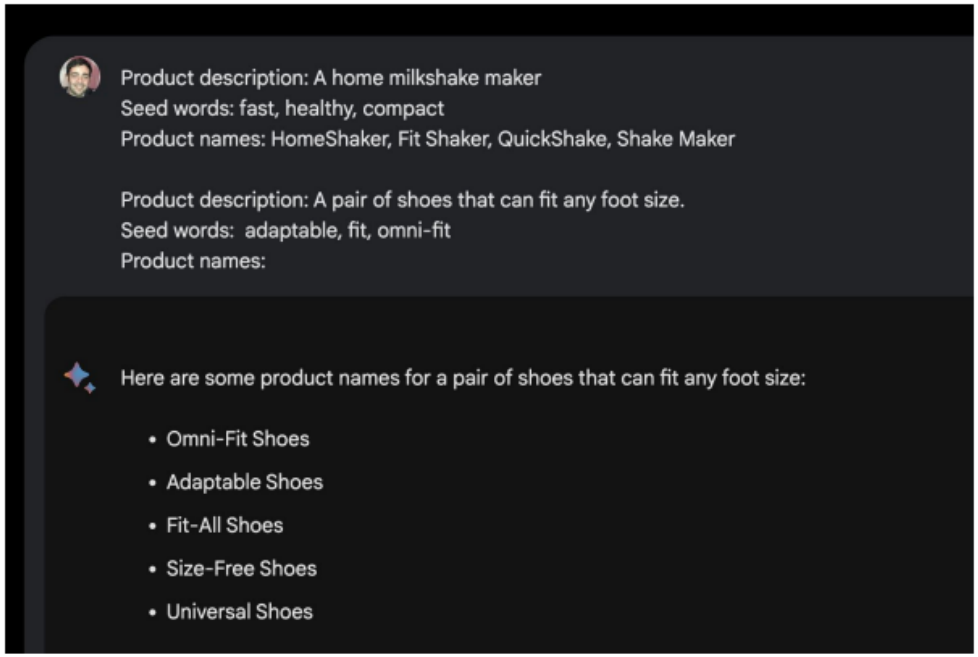


(Ref: The Complete Prompt Engineering for AI Bootcamp (2023))

Created by OpenAI Access it with code or without (Playground https://platform.openai.com/playgro

YHK

Bard — Palm 2/Gemini



(Ref: The Complete Prompt Engineering for AI Bootcamp (2023))

Created by Google Access it via chat https://bard.google.com/ or encounter it in search results

YHK

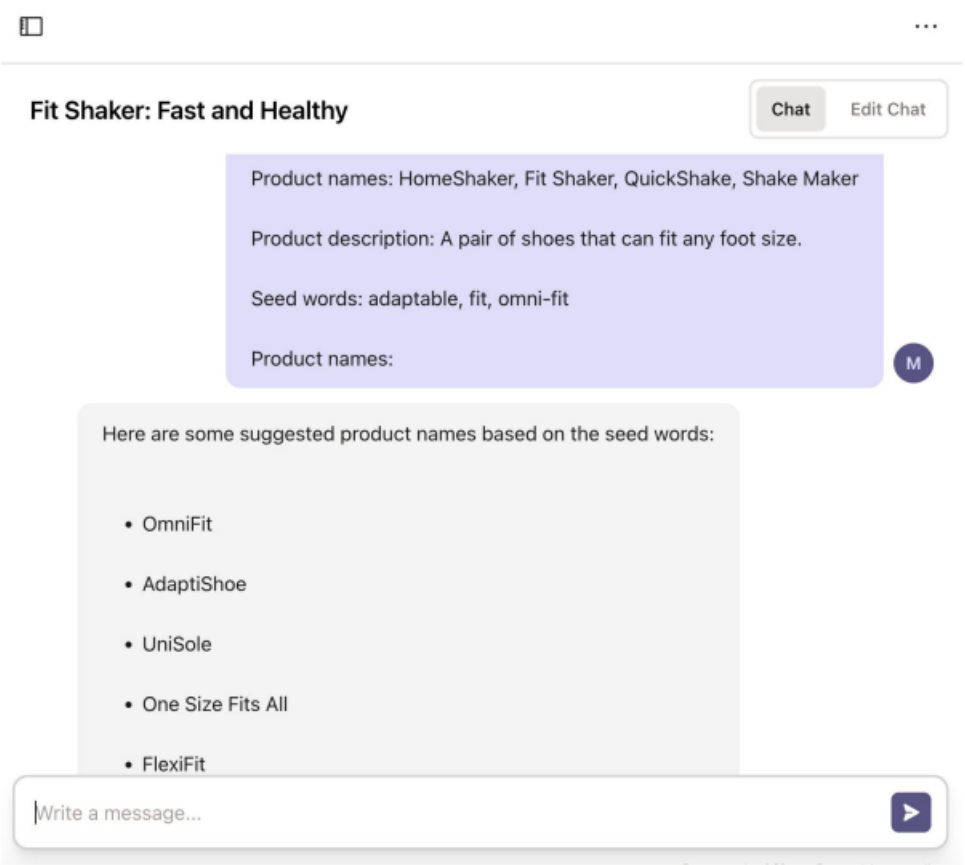
Meta LLaMA

- Open-Source. Need to build a UX and any advanced functionality around it, and may need to fine-tune it.
- Many use-cases in the enterprise can't use OpenAI for fear of sensitive data leaking or being used to train the model (though OpenAI claims to keep API data private).
- If you have 200+ examples fine-tuning beats prompt engineering for a specific defined task.

(Ref: The Complete Prompt Engineering for AI Bootcamp (2023))

YHK

Anthropic Claude

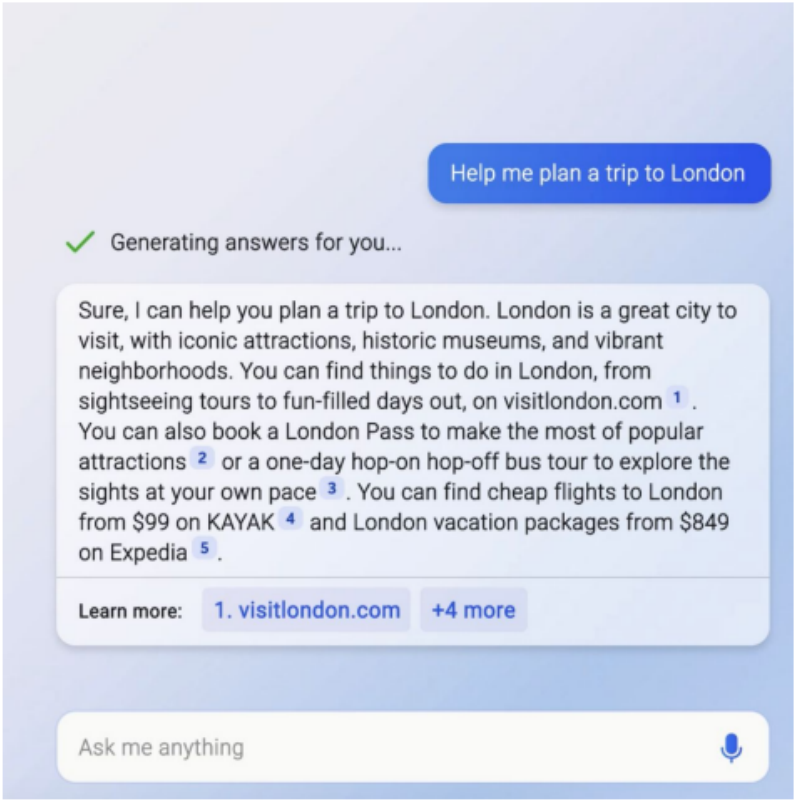


(Ref: The Complete Prompt Engineering for AI Bootcamp (2023))

Created by Anthropic https://console.anthropic.com/ or API Uses Constitutional AI rather than RLHF Constitutional AI trains to follow a set of high-level principles or rules, such as a constitution, that specify the desired behavior and outcomes of the system. RLHF uses human feedback, such as ratings, preferences, or corrections, to optimize a language model or an agent's policy using reinforcement learning

YHK

Microsoft Bing — GPT 4

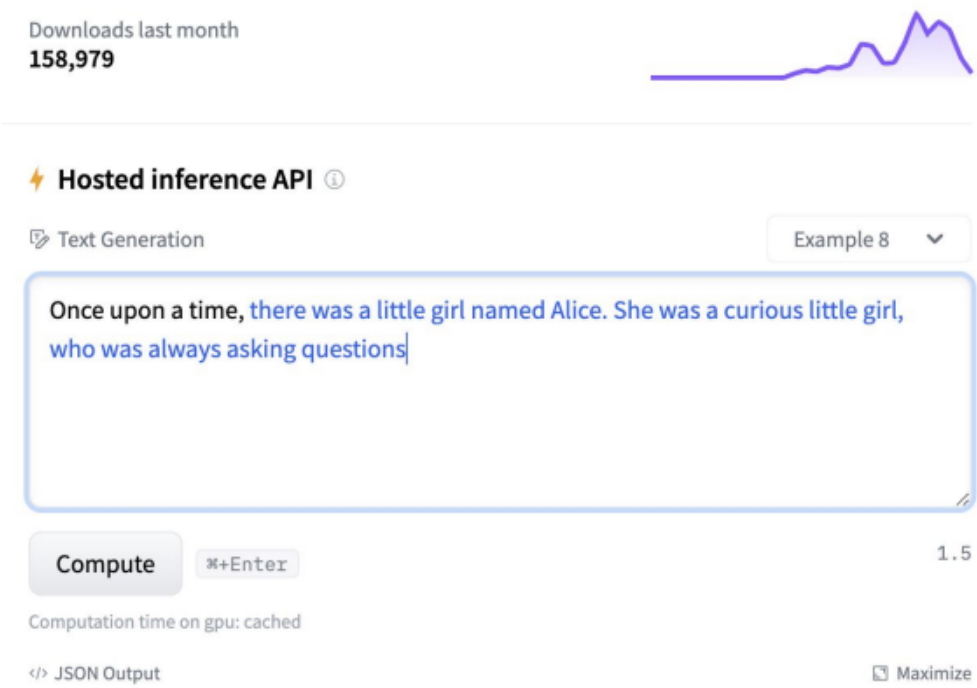


(Ref: The Complete Prompt Engineering for AI Bootcamp (2023))

Powered by OpenAI's GPT-4
https://www.microsoft.com/en-gb/bing

YHK

Falcon



(Ref: The Complete Prompt Engineering for AI Bootcamp (2023))

Access it via HuggingFace transformers library, 7B and 40B models as well as instruct fine-tuned
Features:

- Free for commercial use
- Open source
- Possible to fine-tune

YHK

Leader board (Jan 2024)

Leaders by category

The leaderboard ranks LLMs across multiple prompt categories.

RANK ↑	LLM	TOTAL	BRAINSTORMING	CLOSED QA	GENERATION	OPEN QA	REWRITE
1	GPT-4	81.75	89.74	78.95	81.87	74.10	92.31
2	WizardLM 13B V1.2	79.56	76.92	73.68	80.31	77.11	92.31
3	LLaMA 2 70B Chat	78.97	88.46	68.42	81.35	69.88	84.62
4	GPT-3.5 Turbo	76.79	73.08	68.42	80.31	73.49	76.92
5	Vicuna 33B V1.3	74.21	82.05	47.37	71.50	70.48	76.92
6	Guanaco 13B	50.00	50.00	50.00	50.00	50.00	50.00

(Ref: https://toloka.ai/llm-leaderboard/)

YHK

Want to give it a try? - Hugging Face APIs

(Ref: What are Large Language Models(LLMs)? -Suvojit Hore)

YHK

Sentence Completion

```
import requests
from pprint import pprint

API_URL = 'https://api-inference.huggingface.co/models/bigscience/bloomz'
headers = {'Authorization': 'Entertheaccesskeyhere'}

def query(payload):
    response = requests.post(API_URL, headers=headers, json=payload)
    return response.json()

params = {'max_length': 200, 'top_k': 10, 'temperature': 2.5}
output = query({
    'inputs': 'Sherlock Holmes is a',
    'parameters': params,
})

print(output)

[{'generated_text': 'Sherlock Holmes is a private investigator whose cases '
                    'have inspired several film productions'}]
```

YHK

Question Answers

```
API_URL =
     'https://api-inference.huggingface.co/models/deepset/roberta-base-squad2'
headers = {'Authorization': 'Entertheaccesskeyhere'}


def query(payload):
    response = requests.post(API_URL, headers=headers, json=payload)
    return response.json()

params = {'max_length': 200, 'top_k': 10, 'temperature': 2.5}
output = query({
    'inputs': {
            "question": "What's my profession?",
            "context": "My name is Yogesh and I am an AI Coach"
        },
    'parameters': params
})

pprint(output)

{'answer': 'AI Coach',
 'end': 39,
 'score': 0.7751647233963013,
 'start': 30}
```

YHK

Summarization

```
API_URL = "https://api-inference.huggingface.co/models/facebook/bart-large-cnn"
headers = {'Authorization': 'Entertheaccesskeyhere'}

def query(payload):
    response = requests.post(API_URL, headers=headers, json=payload)
    return response.json()

params = {'do_sample': False}

full_text = '''AI applications are summarizing articles, writing stories and
engaging in long conversations and large language models are doing
the heavy lifting.

:
'''

output = query({
    'inputs': full_text,
    'parameters': params
})
print(output)

[{'summary_text': 'Large language models - most successful '
                  'applications of transformer models. ...'}]
```

YHK

Conclusions, Cautions and What's Next?

So, What are LLMs?

Large Language Models (LLMs) have revolutionized natural language processing, ushering in advancements in text generation and understanding. Key attributes include:

- **Learning from Extensive Data:** LLMs acquire knowledge from vast datasets, resembling a massive library of information.
- **Grasping Context and Entities:** These models understand context and entities, allowing for a deeper comprehension of language.
- **Proficient User Query Responses:** LLMs excel in responding to user queries, showcasing their ability to apply learned knowledge effectively.

Despite their versatile applications across industries, ethical concerns and potential biases necessitate a critical evaluation to understand their societal impact.

YHK

Core Beliefs of Large Language Models

- No inherent "core beliefs."
- Word guessers predicting internet-like sentences.
- Can write both for and against a topic without belief.
- Emulates the most common response in training data.

YHK

Truth and Morality in Large Language Models

- Lack sense of truth or morality.
- Tendency to generate words we agree are true.
- No guarantee of providing the actual truth.

YHK

Mistakes in Large Language Models

- Prone to mistakes due to inconsistent training data.
- Self-attention may not capture all relevant information.
- Hallucination: generating words not derived from input.
- Preference for common words, small numbers, and specific names.

YHK

Auto-regressive Nature of LLMs

- Auto-regressive models: guesses affect subsequent inputs.
- Errors accumulate, potentially compounding mistakes.
- No mechanism to "change minds" or self-correct.
- Lack the ability to retry or undo prior choices.

YHK

Verification of Outputs

- Always verify outputs of large language models.
- Assess competence to verify results in high-stakes tasks.
- Mistakes in critical tasks may lead to costly decisions.

YHK

Input Size and Memory Limitations

- Large language models have input size limits.
- Conversation appears coherent until log size exceeds limit.
- Earlier parts of the conversation are deleted, and the model "forgets."

YHK

Ethical Considerations

- Awareness of potential biases in LLMs is crucial for responsible usage.
- Continuous evaluation of ethical implications is necessary to mitigate societal risks.
- Balancing the benefits of LLMs with ethical concerns ensures responsible deployment.

YHK

Future Impact

- LLMs expected to revolutionize domains such as job markets, communication, and society.
- Careful use and ongoing development are essential for positive impacts.
- Understanding limitations and ethical considerations is vital for responsible integration into various domains.

YHK

Landscape of LLMs & Quiz

- Types of models - Foundation models, LLM, SLM, VLMs, etc.
- Common LLM terms - Prompts, Temperature, Hallucinations, Tokens, etc.
- LLM lifecycle stages - Pre-training, Supervised Fine Tuning, RLHF, etc.
- LLM evaluations - ROUGE, BLEU, BIG-bench, GLUE, etc.
- LLM architecture - Encoder, Decoder, Transformer, Attention, etc.
- Retrieval augmented generation - Vector DBs, Chunking, Evaluations, etc.
- LLM agents - Memory, Planning, ReAct, CoT, ToT, etc.
- Cost efficiency - GPU, PEFT, LoRA, Quantization, etc.
- LLM security - Prompt Injection, Data poisoning, etc.
- Deployment & inference - Pruning, Distillation, Flash Attention, etc.
- Platforms supporting LLMOps

(Ref: LinkedIn post by Abhinav Kimothi - 23 Jan 2024)

YHK

Conclusions

Conclusions

- Robust NLP preprocessing and embeddings are critical for effective information retrieval.
- spaCy and scikit-learn streamline NLP + ML workflows.
- Vector-based understanding enables semantic and contextual search.
- Combining these components forms the backbone of Retrieval-Augmented Generation.
- Future directions:
  - Fine-tuning LLMs for domain tasks
  - Efficient vector database retrieval (FAISS, Milvus)
  - Integrating sentiment and entity-level reasoning

YHK

References

- Explosion AI, *spaCy Documentation*, `https://spacy.io/`
- Pedregosa et al., *Scikit-learn: Machine Learning in Python*, JMLR, 2011.
- Mikolov et al., *Efficient Estimation of Word Representations in Vector Space*, arXiv, 2013.
- Bird et al., *Natural Language Processing with Python*, O'Reilly, 2009.
- OpenAI, *Retrieval-Augmented Generation (RAG) Overview*, 2024.
- Manning & Schütze, *Foundations of Statistical Natural Language Processing*, MIT Press.
- Jurafsky & Martin, *Speech and Language Processing*, 3rd Ed. Draft, 2023.

YHK

Thanks ...

- Office Hours: Saturdays, 3 to 5 pm (IST); Free-Open to all; email for appointment to yogeshkulkarni at yahoo dot com
- Call + 9 1 9 8 9 0 2 5 1 4 0 6

(https://www.linkedin.com/in/yogeshkulkarni/)

(https://medium.com/@yogeshharibhaukulkarni)

(https://www.github.com/yogeshhk/)

YHK