Advanced Information Retrieval

Introduction

TU WIEN

AIR – 2021

Our Online Format

- Weekly YouTube uploads of recorded lectures
  - 45min to 1 hour each
  - Additionally: PDF slides + automatic closed caption text
- Flexible grading structure
- Weekly online office hours for exercises & lectures
- 24h take home exam (2 dates offered)

Problems / Questions / Feedback

Use the TUWEL Forum

or write an email to sebastian.hofstaetter@tuwien.ac.at

Google

information retrieval

All | Images | Videos | News | Books | More | Settings | Tools

About 46.400.000 results (0,43 seconds)

Dictionary

Enter a word, e.g. 'pie'

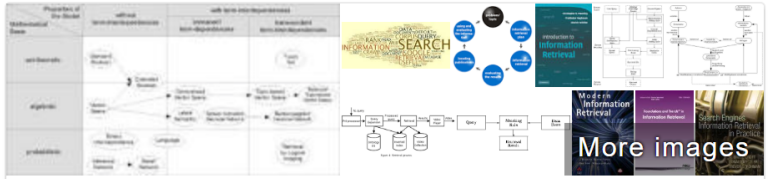
information retrieval

noun COMPUTING

the tracing and recovery of specific information from stored data.
"an information retrieval system"

Translations, word origin, and more definitions

Feedback



Information retrieval

Information retrieval is the activity of obtaining information system resources relevant to an information need from a collection of information resources. Searches can be based on full-text or other content-based indexing. Wikipedia

Feedback

Information retrieval - Wikipedia

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

Information retrieval (IR) is the activity of obtaining **information** system resources relevant to an **information** need from a collection of **information** resources. Searches can be based on full-text or other content-based indexing.

Overview · History · Model types · Timeline

Information Retrieval – Wikipedia

https://de.wikipedia.org/wiki/Information_Retrieval Translate this page

Information Retrieval [ˌɪnfəˈmeɪʃən rɪˈtriːvəl] (**IR**) oder Informationsrückgewinnung, gelegentlich ungenau Informationsbeschaffung, ist ein Fachgebiet, ...

Geschichte · Grundbegriffe · Relevanz und Pertinenz · Typologie von ...

[PDF] Introduction to Information Retrieval - Stanford NLP Group

https://nlp.stanford.edu/IR-book/pdf/01bool.pdf

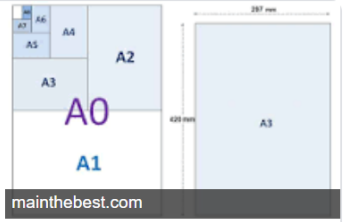
Information retrieval (IR) is finding material (usually documents) of an unstructured nature (usually text) that satisfies an **information** need from within large collections (usually stored on computers).

Information Retrieval

a3 size

All Images Maps News Videos More Settings Tools

About 289.000.000 results (0,73 seconds)



21.0 x 29.7cm

The **A3 size** print measures 29.7 x 42.0cm, 11.69 x 16.53 inches, if mounted 40.6 x 50.8cm, 15.98 x 20 inches. The A4 **size** print measures 21.0 x 29.7cm, 8.27 x 11.69 inches, if mounted 30.3 x 40.6cm, 11.93 x 15.98 inches.

Paper Sizes A0, A1, A2, A3, A4 - Stephen Wiltshire
https://www.stephenwiltshire.co.uk › paper_sizes

About Featured Snippets Feedback

Google.com – **14.10.2019**

a3 size

All Images Maps News Videos More Settings Tools

About 646.000.000 results (0,50 seconds)

29.7 x 42.0cm

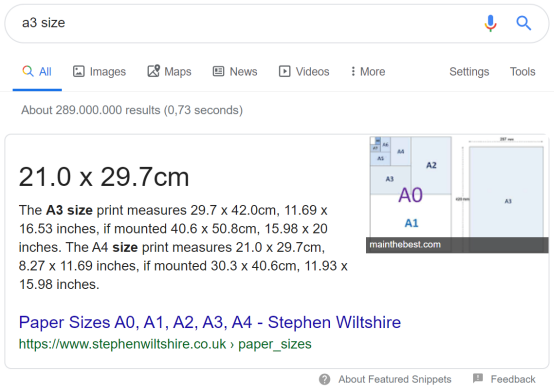
The A3 size print measures **29.7 x 42.0cm**, 11.69 x 16.53 inches, if mounted 40.6 x 50.8cm, 15.98 x 20 inches. The A4 size print measures **21.0 x 29.7cm**, 8.27 x 11.69 inches, if mounted 30.3 x 40.6cm, 11.93 x 15.98 inches.

www.stephenwiltshire.co.uk › paper_sizes
Paper Sizes A0, A1, A2, A3, A4 - Stephen Wiltshire

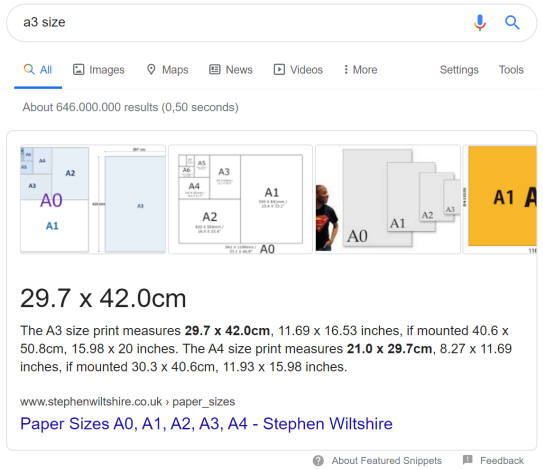
About Featured Snippets Feedback

Google.com – **3.3.2020**

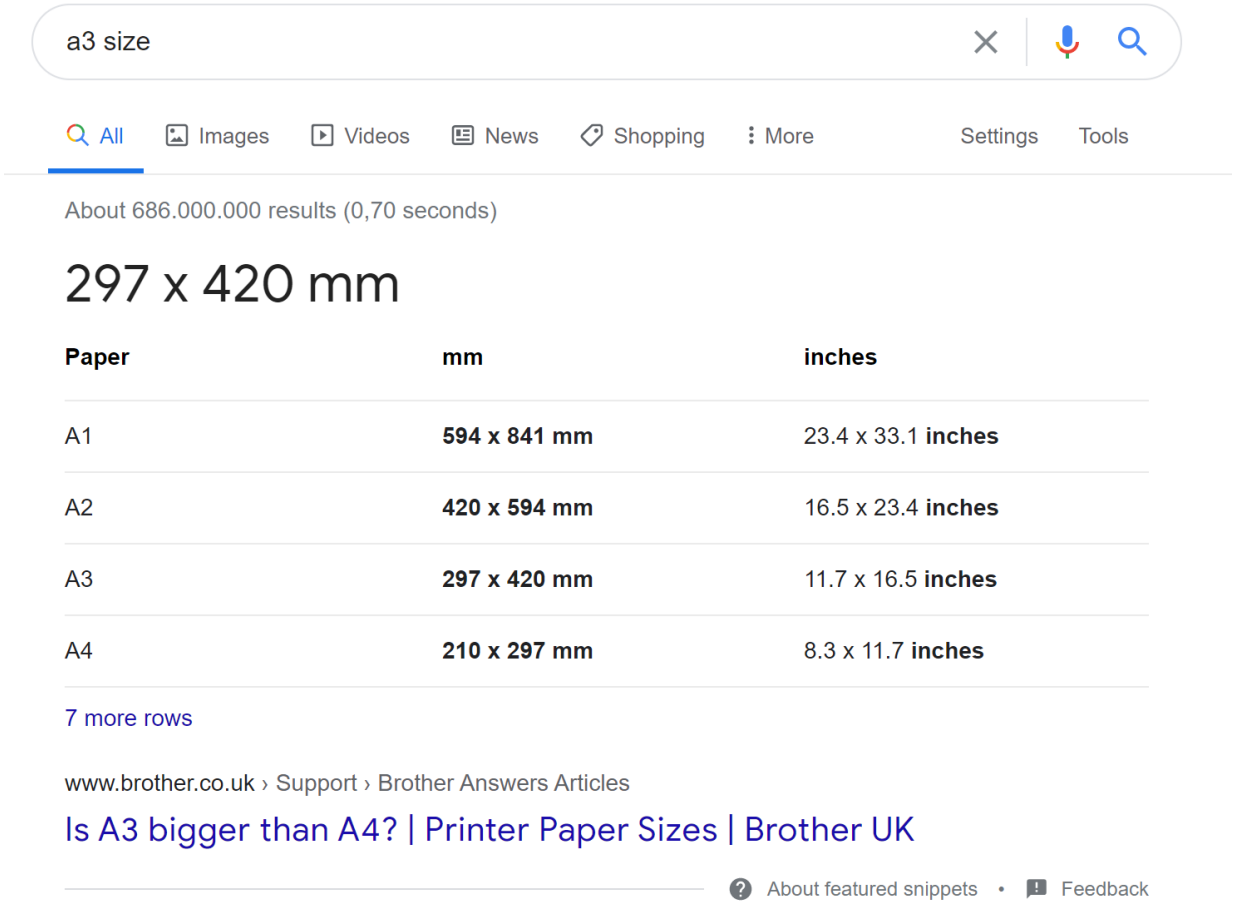
Information Retrieval



Google.com – **14.10.2019**



Google.com – **3.3.2020**



Google.com – **11.2.2021**

Machine Learning

Mat Velloso @matvelloso · 22 Nov 2018

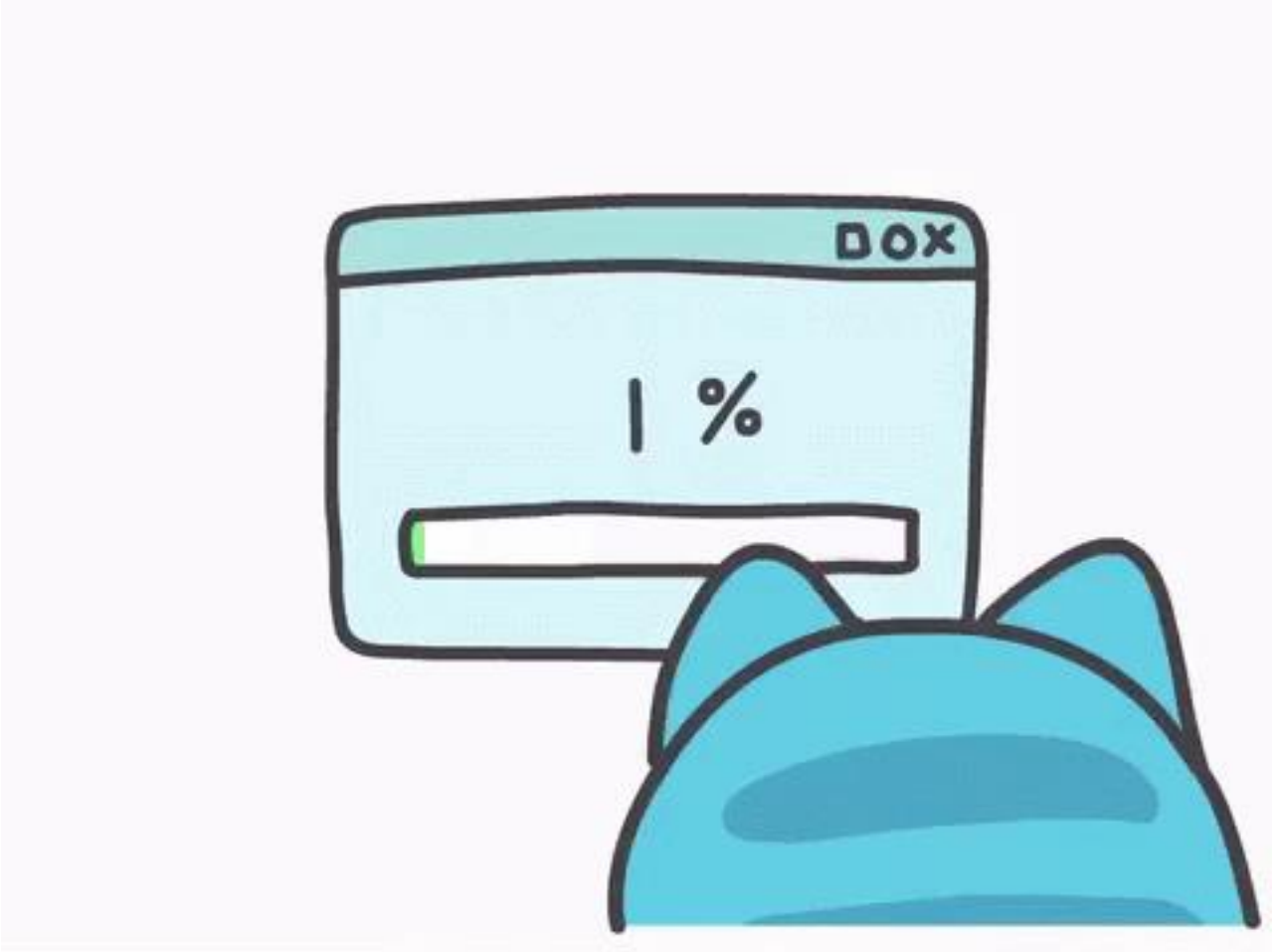
Difference between machine learning and AI:

If it is written in Python, it's probably machine learning

If it is written in PowerPoint, it's probably AI

193 7.9K 22K

Machine Learning



From: https://twitter.com/SwiftOnSecurity/status/969213410490675201

Recommended Prerequisites

- Machine Learning know how
  - Know the basic concepts
  - Experience with a neural network framework (PyTorch, TensorFlow, etc…)
  - Experience in reading academic papers
- Basic IR course
  - Always good, but we will revisit the basics
- Good programming skills
- Available Nvidia GPU (or alternatively a free GPU from Google Collab)

Some pointers to get you started …

- Neural Network Methods in Natural Language Processing by Yoav Goldberg
  - Contains a good introduction to ML as well

- Pretrained Transformers for Text Ranking: BERT and Beyond by Lin et al. https://arxiv.org/abs/2010.06467
  - Survey of neural IR progress starting in 2019

- Google crash course on ML https://developers.google.com/machine-learning/crash-course/ml-intro
- PyTorch Tutorials https://pytorch.org/tutorials/
- AllenNLP Tutorials https://allennlp.org/tutorials

Organization

Lectures, Exercises, Grading

Syllabus

❶ Crash Course IR

- **Fundamentals:** Inverted index & probabilistic scoring (BM25)
- **Evaluation:** List-based measures (binary & graded relevance)
- **Test Collections:** Create and analyze IR datasets

❷ Representation Learning (NLP)

- **Word Embeddings:** Basic building blocks & intro to vector representations
- **Sequence Representations:** Contextual vectors with: CNNs, RNNs, & (pre-trained) Transformers
- **Extractive QA:** Find answer location in text

❸ Neural IR

- **Re-ranking:** From early beginnings of neural re-ranking to pre-trained BERT
  - **From scratch:** early IR specific re-ranking models
  - **Efficient Transformers**: Transformer-Kernel family
  - **State-of-the-art**: Large BERT-based models
- **Domain-Specific:** Caveats and task-changes between passage/document + web and legal/patent domains
- **Retrieval:** Encoding passages into single vectors; directly retrievable with embedded query rep from nearest neighbor index
- **Knowledge Distillation:** Improving the training of efficient architectures with the help of slow, but good models

Syllabus

❶ Crash Course IR

- **Fundamentals:** Inverted index & probabilistic scoring (BM25)
- **Evaluation:** List-based measures (binary & graded relevance)
- **Test Collections:** Create and analyze IR datasets

❷ Representation Learning (NLP)

- **Word Embeddings:** Basic building blocks & intro to vector representations
- **Sequence Representations:** Contextual vectors with: CNNs, RNNs, & (pre-trained) Transformers
- **Extractive QA:** Find answer location in text

❸ Neural IR

- **Re-ranking:** From early beginnings of neural re-ranking to pre-trained BERT
  - **From scratch:** early IR specific re-ranking models
  - **Efficient Transformers**: Transformer-Kernel family
  - **State-of-the-art**: Large BERT-based models
- **Domain-Specific:** Caveats and task-changes between passage/document + web and legal/patent domains
- **Retrieval:** Encoding passages into single vectors; directly retrievable with embedded query rep from nearest neighbor index
- **Knowledge Distillation:** Improving the training of efficient architectures with the help of slow, but good models

Syllabus

❶ Crash Course IR

- **Fundamentals:** Inverted index & probabilistic scoring (BM25)
- **Evaluation:** List-based measures (binary & graded relevance)
- **Test Collections:** Create and analyze IR datasets

❷ Representation Learning (NLP)

- **Word Embeddings:** Basic building blocks & intro to vector representations
- **Sequence Representations:** Contextual vectors with: CNNs, RNNs, & (pre-trained) Transformers
- **Extractive QA:** Find answer location in text

❸ Neural IR

- **Re-ranking:** From early beginnings of neural re-ranking to pre-trained BERT
  - **From scratch:** early IR specific re-ranking models
  - **Efficient Transformers**: Transformer-Kernel family
  - **State-of-the-art**: Large BERT-based models
- **Domain-Specific:** Caveats and task-changes between passage/document + web and legal/patent domains
- **Retrieval:** Encoding passages into single vectors; directly retrievable with embedded query rep from nearest neighbor index
- **Knowledge Distillation:** Improving the training of efficient architectures with the help of slow, but good models

Lectures / Content

- Find extra information here:
  - Introduction to IR slides
  - Additional lecture notes

github.com/sebastian-hofstaetter/teaching

- Star it + add your content via issues & pull requests (🎁 bonus points!)
  - Fix automatic closed captions (5 bonus points for each lecture)
  - Lecture notes, summaries, examples, bugfixes … (generous bonus points)

Exercises

❶ Data annotation

- Understand the task, that we want to teach the machine
- Create testing & analysis data for exercise 2

❷ Neural Re-ranking & Extractive QA

- Using Python & PyTorch
- **Part 1:**
  - Implement & train neural re-ranking models from scratch
- **Part 2:**
  - Use pre-trained models from HuggingFace to create an extractive QA system

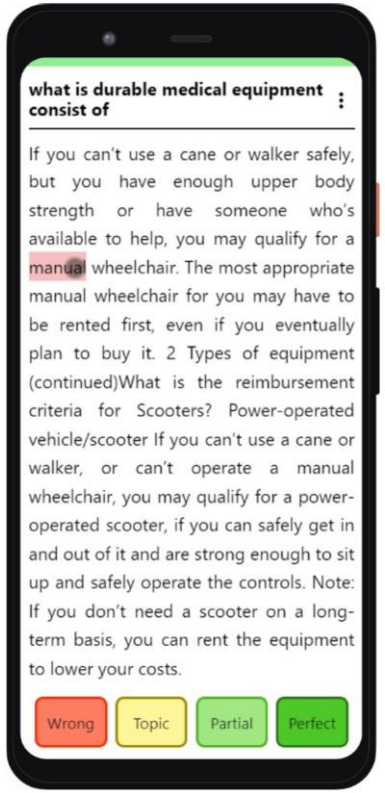
Exercise 1

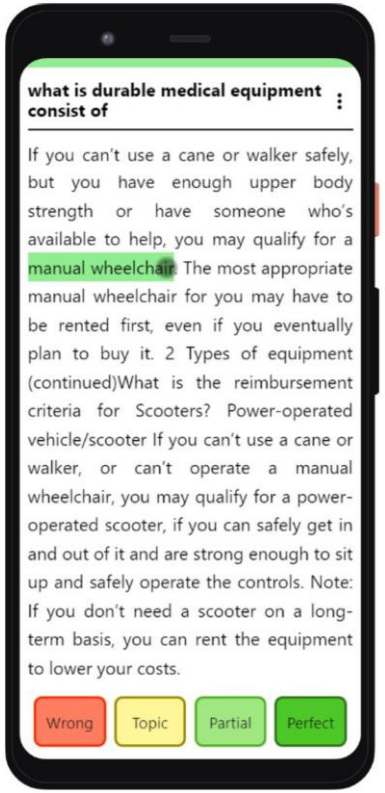
- Creating annotations is time consuming;
  easier to split the task among many people
  - Each student spends a few hours (500 annotations for 100%, 250 min.)
- We create a fine-grained passage retrieval and extractive QA dataset, based on MSMARCO
  - This fine-grained data over a lot of queries doesn’t exist yet
- We use this dataset in Exercise 2 for evaluation and exploration
  - Potentially we also share it with the research community
  - Data is completely anonymized before publication

Exercise 1 - FiRA

- We created a specialized & simple to use tool (FiRA) for mobile and desktop use
- Each registered student receives pre-created account information via email

- More annotations = bonus points
  - +4% of the total grade per 100 extra annotations
  - > 1.000 annotations remove min. point requirement of exercise 2 & exam





Exercise 2

- 2 Parts (both must be done):

❶ Implement & train neural re-ranking models from scratch

- Not state of the art, but teaches you how PyTorch works from the ground up and allows you to learn to work with training loops, loss functions, and tensor operations

❷ Use pre-trained models from HuggingFace to create an extractive QA system

- Now, we don’t train a model, but download a pre-trained model; and put together the pipeline necessary to get google-like results (example from the beginning)

Exercise 2

- Exercise 2 in groups
  - 3 persons per group (managed via TUWEL)
  - Will be evaluated together
- Work in 1 private GitHub repository (via GitHub classroom)
- Lots of bonus point opportunities 🎉👏💯✨
  - For creative extra work, if you have fun doing it, go for it and you'll get points!
  - For finding & fixing bugs in the starter code or lecture slides

Online Exam

- 24-hour take-home exam
- Type: Paper reading with questions showing the understanding of it
  - You must answer both easier & more complicated questions
  - Relates to one or more lectures
  - Open-Book

- Exam on: **26.5.** & **16.6.** Starting at noon (12:00)
  - Pick **one** date (do-overs only for failed attempts or technical problems)
  - Administered completely via TUWEL Test

Grading

Exercise 1 (Annotation):	10%	(min 50% to pass)
Exercise 2 (Neural IR):	50%	(min 50% to pass)*
Exam:	40%	(min 30% to pass)*
Total	**100%**	**(min 50% to pass)**

Grading Scheme (TUWEL defaults)

1: >= 88,00 2: >= 75,00 3: >= 63,00 4: >= 50,00

* Can be removed with enough bonus points in Exercise 1

See you next week – virtually 👋

For feedback, problems, or any other issues, please write to:
sebastian.hofstaetter@tuwien.ac.at