Web and Social Media Analytics

Search engines – part I

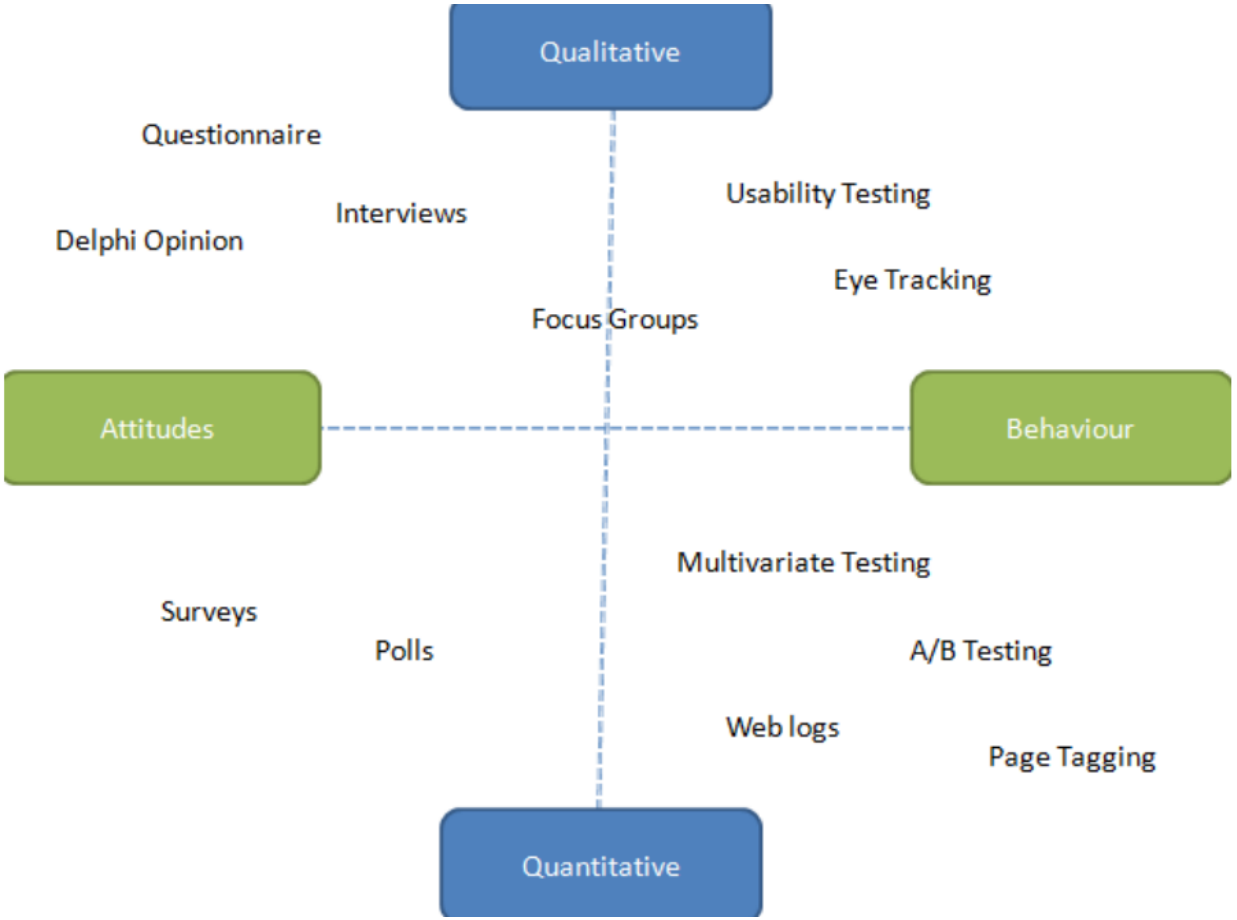
Dr Philip Worrall

School of Computer Science and Engineering
115 New Cavendish Street
University of Westminster
London, W1W 6UW
worralph@westminster.ac.uk

LW3

With acknowledgement and thanks to Vassiliki Bouki

Recap: data sources



Recap: The web analytics process

1. Identification of organisational **issues**
2. Develop an understanding of the organisation's **purpose, strategy** and **goals**
3. Determine what **measures of success** might be relevant and convert them into **web metrics**
4. Implement **data collection** and any required tools
5. Use the longer term-aims and objectives of the organisation to create shorter term **Key Performance Indicators** (KPI)
6. Monitor these KPIs over time. Make **decisions** and take **actions**.
7. Re-evaluate **progress** and make appropriate **adjustments**.

Recap: KPIs

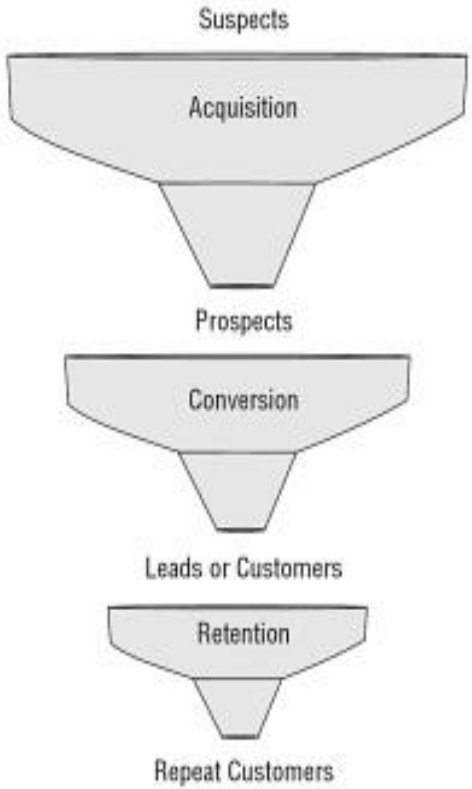
#1 KPI Name (author, date created)

	RAW	PROGRESS	CHANGE
Example	10 new users	10% of all visitors	+10% more donations
Source of data	Does the data already exist or do we need to collect it? How far back does the data go? Is there an existing Google Analytics report that we can use? Is the data stored locally?		
Frequency	Daily? Weekly? Monthly? Quarterly? Yearly? How often should we monitor our progress?		
Target	What is the target we are trying to hit? How can we track progress graphically? All KPIs must be temporal. What is the deadline?		
Alignment	How does the KPI relate to the organisation's long-term strategy and goals?		

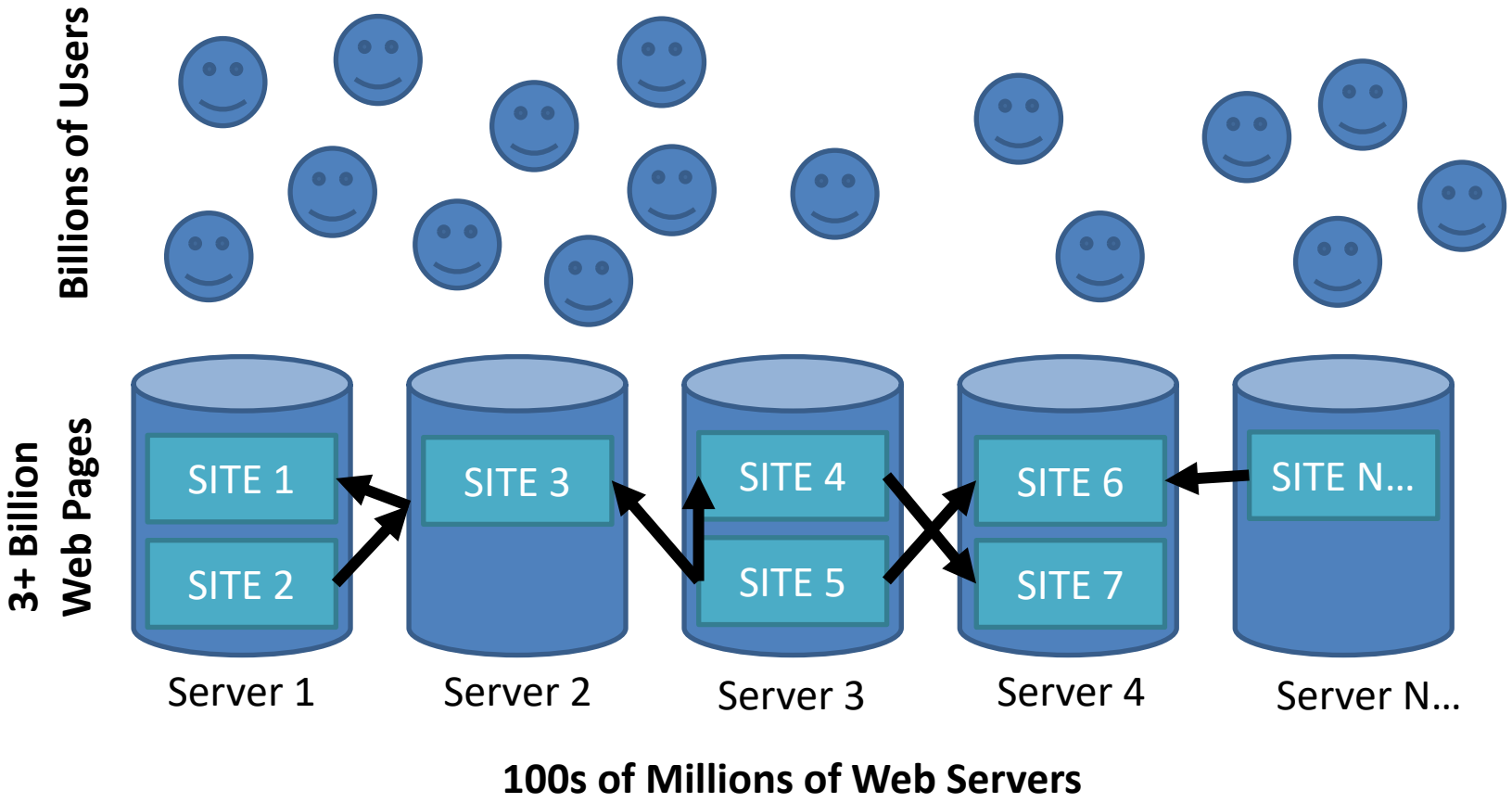
Week 3

Plan for today

- Information and data retrieval systems (IR, DR)
  - Retrieval models
- Search engines (SE)
  - Their relation to acquisition in Ash's (2012) marketing funnel



Size and Layout of the web



Locating relevant content

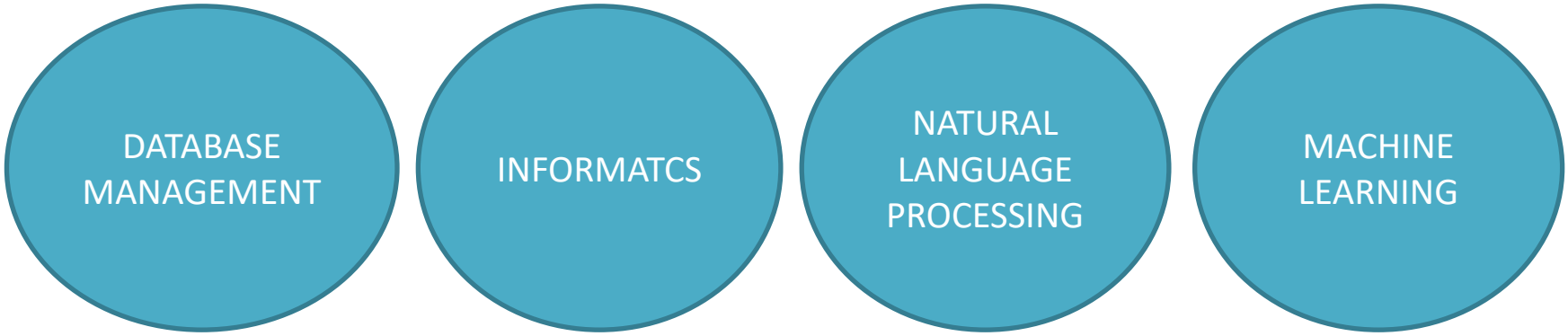
- **In earlier versions of the web, the primary way to locate relevant information was through hyperlinks**
- As the web grew, alternative methods of using web directories were proposed.
  - Each maintainer of a directory curated links to relevant sites in their niche.
- A longer term solution was needed.
- Throughout the 1990s and early 2000s several different search engines entered the market using ideas from information retrieval (IR) theory.

Information Retrieval

Information retrieval (IR) systems

- Any system that enables users to obtain information (**relevant documents or fragments**) relevant to an information need (**query**) from a document collection (**corpus**). Searches can be based on metadata or on full-text (or other content-based) indexing.

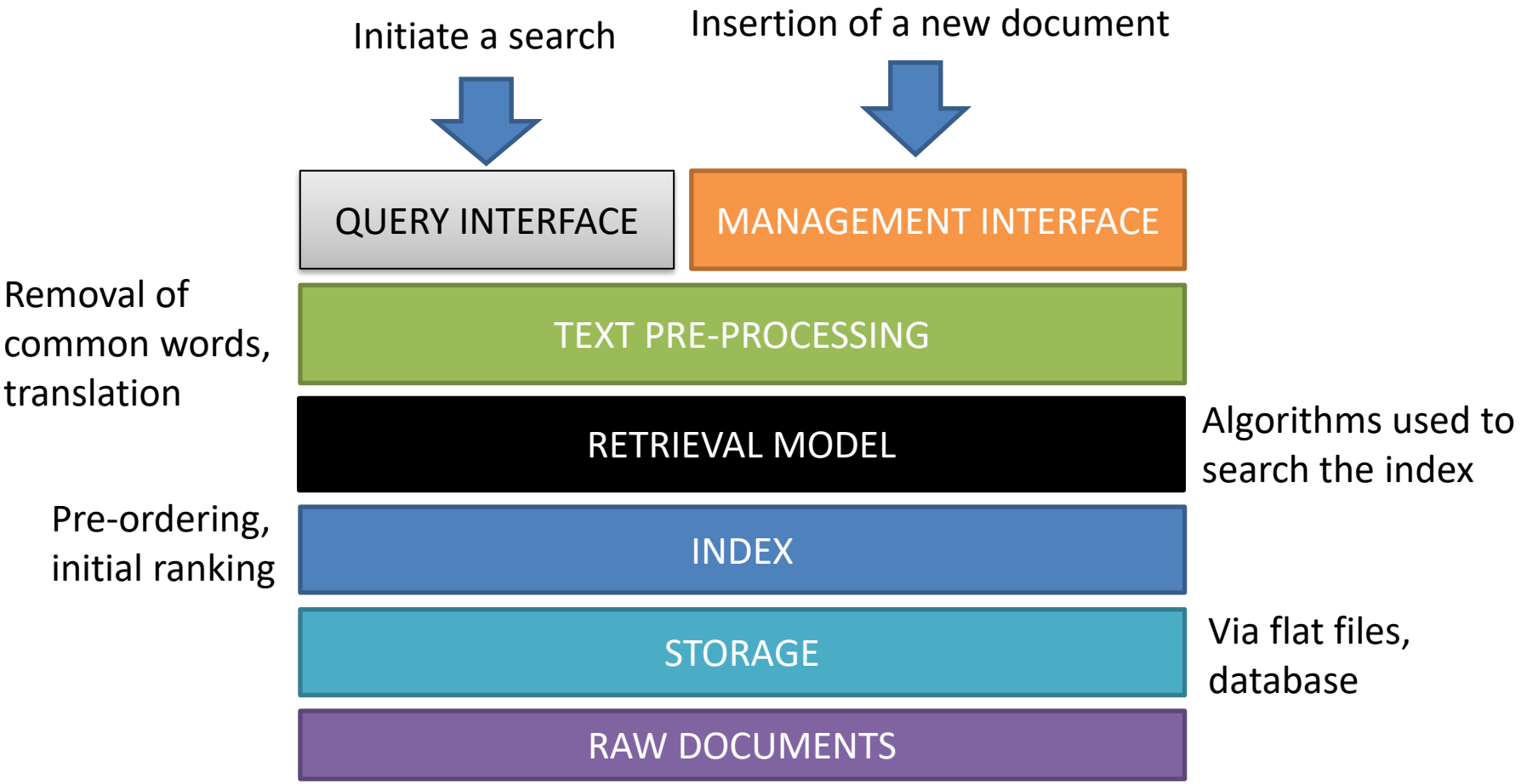
IR Systems and theory involve a number of specialist scientific disciplines



Data vs Information retrieval

	Data Retrieval	Information Retrieval
What we retrieve	Structured Data / Clear Semantics	Mostly unstructured / free text with metadata
Queries	Formally defined queries/ unambiguous	Vague / ambiguous / often expressed in Natural Language
Results	Exact	Sometimes relevant – others not relevant
Interaction	Ad hoc, reporting	Significant interaction, research, refinement of query

Architecture of an IR system



The indexing component

RAW DOCUMENT

#1 “The hotel is located next to the city railway station.

RAW DOCUMENT

#2 The Plaza hotel is a 2 min walk from the coastline.

INDEX

Organised Alphabetically

TERM	DOCUMENT	POSITION
city	#1	8
coastline	#2	11
from	#2	9
hotel	#1	2
hotel	#2	3

The term “a” doesn’t appear in the index even though it would be placed before “city”. **WHY?**

The text pre-processing component

TEXT PRE-PROCESSING

- Transforms USER **queries** before they are checked against the **INDEX**
- Modifies the **TERMS** placed into the index when a new **DOCUMENT** is added to the collection.

WHY?

- Which documents would the TERM “City” match?
- How about the TERMS “London”, “Madrid” or “hotels”?
- What we do we know about the TERMS “a”, “and”, “to”, “in”.

Text pre-processing strategies

TEXT PRE-PROCESSING

- **Tokenisation**: break text into individual TERMS (keywords)
- **Stemming** and **lemmatisation**:
  - TERMS to 'root' words
  - e.g. Computer -> comput, running -> ran
- **Stopwords** removal: e.g. "a", "the", "it", "to"
- **Strip unwanted characters**/markup:
  - HTML tags, punctuation, symbols, numbers, etc.
- Detect **common phrases** and **named entities:**
  - Places, people, countries, companies

Retrieval models and algorithms

RETRIEVAL MODEL

- When a QUERY is conducted, documents in the collection will fall into one of **FOUR** possible categories.
  1) Relevant documents that are selected
  2) Relevant documents that are not selected
  3) Non-Relevant documents that are selected
  4) Non-Relevant documents that are not selected.

It is the responsibility of the retrieval model (algorithm) to search the INDEX to identify relevant documents and determine a possible ranking.

Measuring performance

RETRIEVAL MODEL

- $$Precision = \frac{\text{relevant and selected}}{\text{relevant and selected + non}-\text{relevant and selected}}$$

- $$Recall = \frac{\text{relevant and selected}}{\text{relevant and selected + relevant and not selected}}$$

Precision relates to the retrieval models ability to deliver a high proportion of results that the user will find useful.

Recall reflects the ability of the retrieval model to correctly identify relevant documents in the collection.

Potential algorithms

RETRIEVAL MODEL

Standard Boolean Model

- Documents are selected and retrieved if the search TERM appears in a document.

Term-Frequency Model

- Documents with the highest frequencies of the search TERM are selected.

Term-Frequency-Density Model

- Documents with the highest frequencies of the search TERM relative to their total length are selected.

Potential algorithms

RETRIEVAL MODEL

Probabilistic (statistical) models

- Imagine we have 10 documents in our COLLECTION.
- Each time we receive a QUERY we show the 5 random DOCUMENTS but each time the order of their appearance is randomised.
- *A priori*, what is the expected probability of each document being clicked on?
- Suppose we conducted an experiment where we asked users to use the same query and click the link they thought was most appropriate.
- **What if** DOCUMENT 1 receives 50% of all clicks?

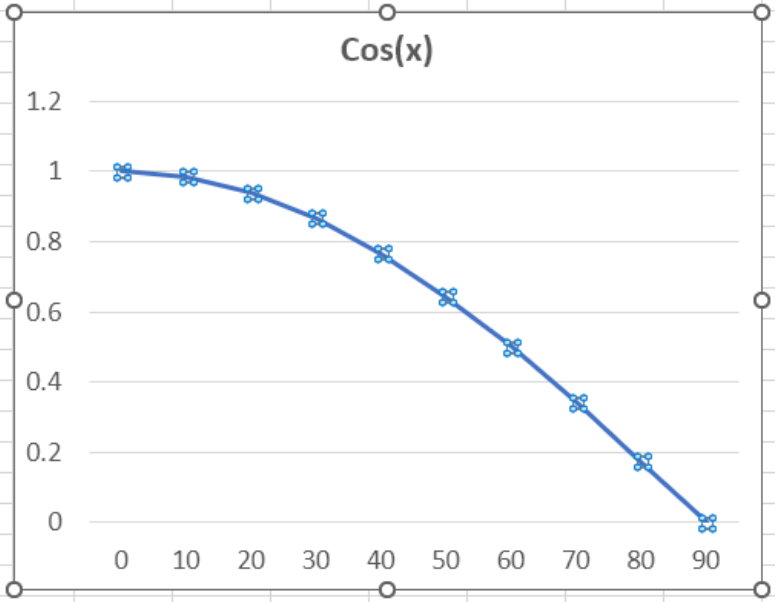
Potential algorithms

RETRIEVAL MODEL

Cosine Similarity

- Measures the similarity of two vectors based on the angle formed between when plotted in number space

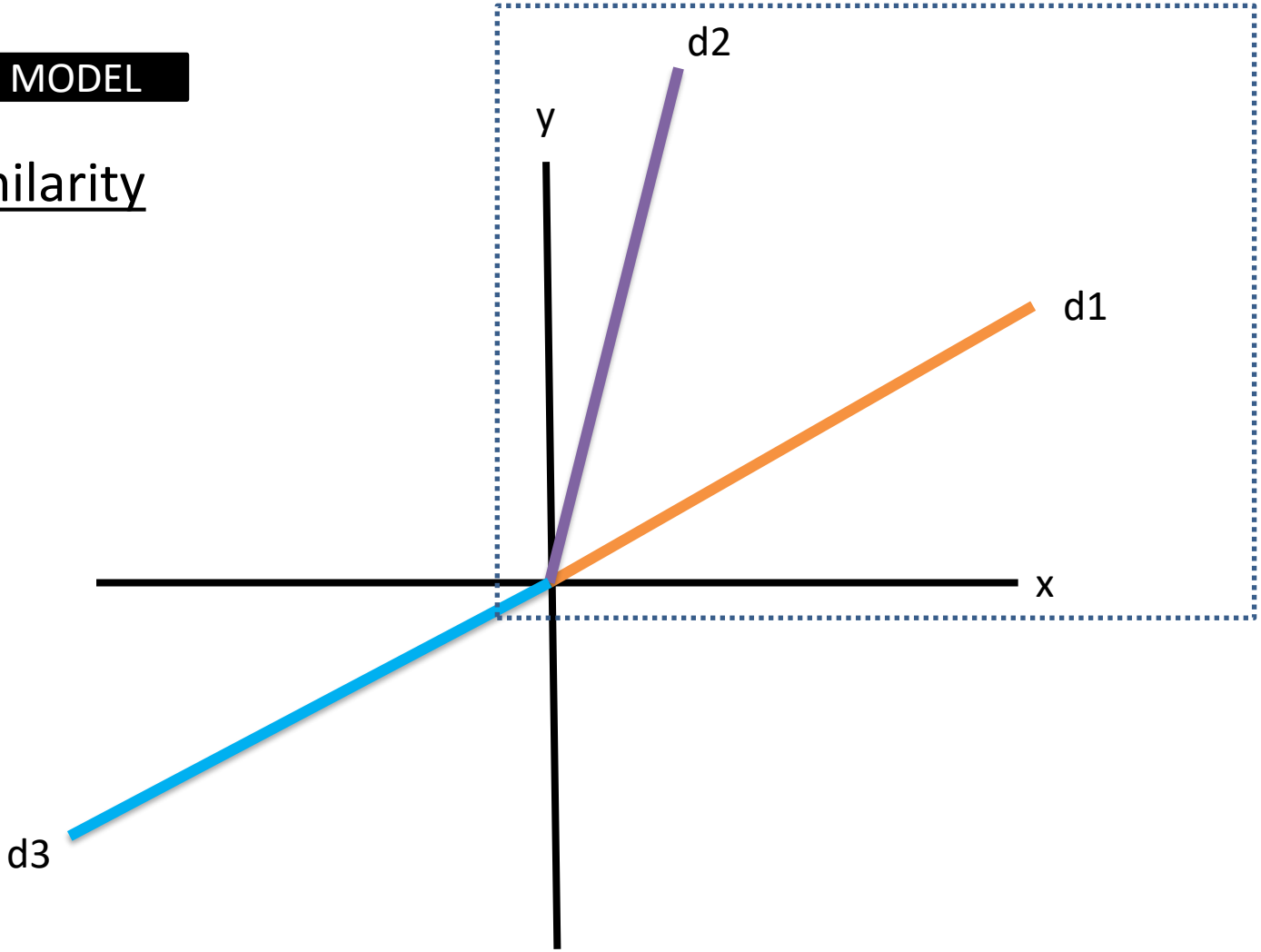
Degrees	Radians	Cos(x)
0	0	1
10	0.174533	0.984807753
20	0.349066	0.939692621
30	0.523599	0.866025404
40	0.698132	0.766044443
50	0.872665	0.64278761
60	1.047198	0.5
70	1.22173	0.342020143
80	1.396263	0.173648178
90	1.570796	6.12574E-17



Potential algorithms

RETRIEVAL MODEL

Cosine Similarity



Potential algorithms

RETRIEVAL MODEL

Cosine Similarity

TF (Augmented)

$$TF(w,d) = 0.5 + \frac{0.5 \cdot f(w,d)}{max(f(w,d) : w \in d)}$$

IDF

$$IDF(w,D) = log \cdot \frac{|D|}{|(d \in D : w \in d)|}$$

- To apply cosine similarity in IR systems we have to represent each document as a vector (sequence of numbers)
- This mapping is commonly done through the application of the ***vector space model*** (TF-IDF).
- **TF** (Term Frequency) increases for documents containing a higher proportion of a term.
- **IDF** (Inverse Document Frequency) penalises documents that use a high proportion of commonly used words.

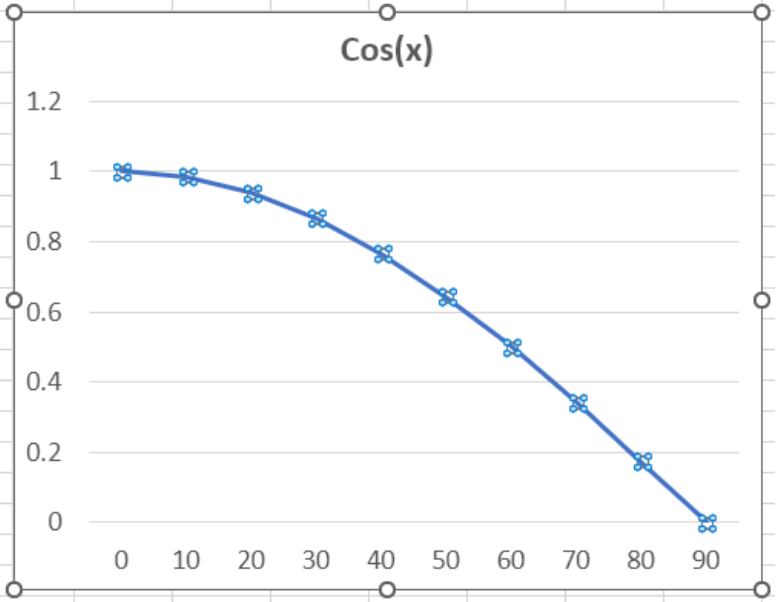
Potential algorithms

RETRIEVAL MODEL

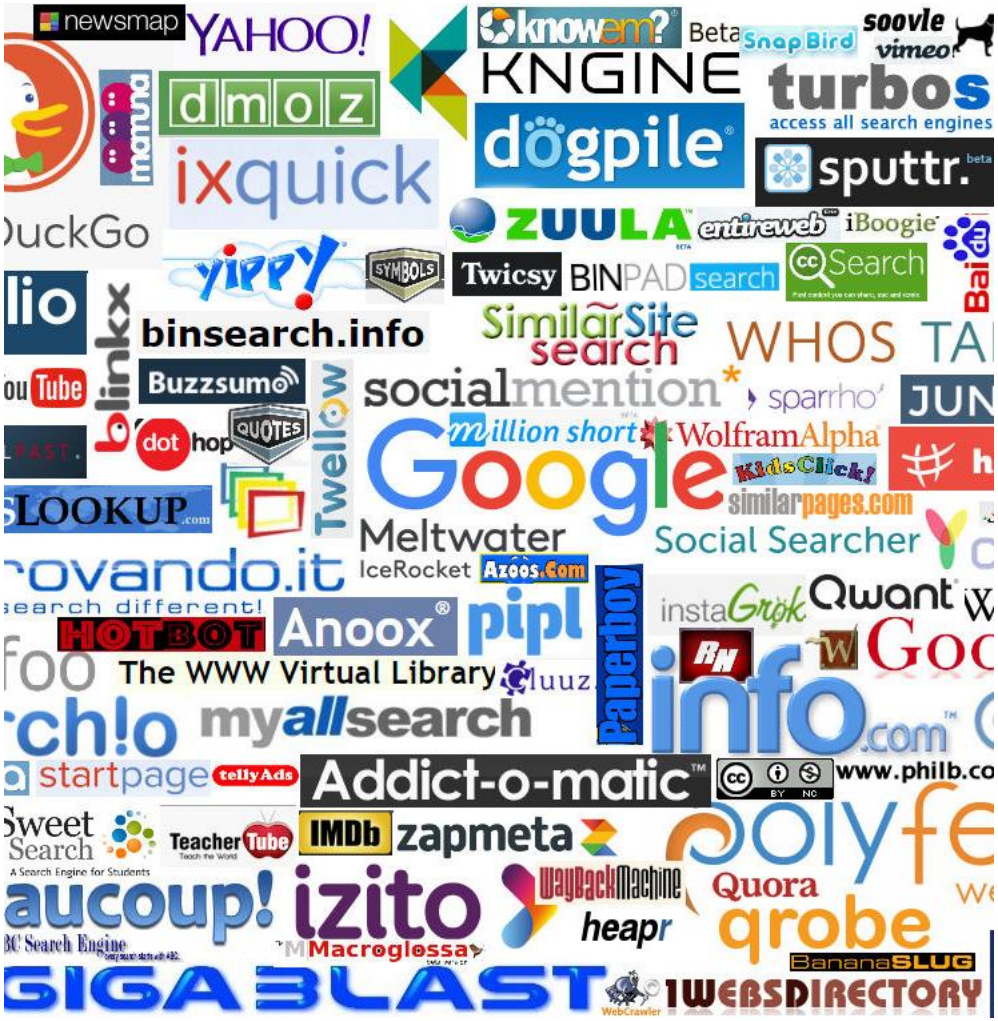
Cosine Similarity

- Measures the similarity of two vectors based on the angle formed between when plotted in number space

Degrees	Radians	Cos(x)
0	0	1
10	0.174533	0.984807753
20	0.349066	0.939692621
30	0.523599	0.866025404
40	0.698132	0.766044443
50	0.872665	0.64278761
60	1.047198	0.5
70	1.22173	0.342020143
80	1.396263	0.173648178
90	1.570796	6.12574E-17



Web search engines



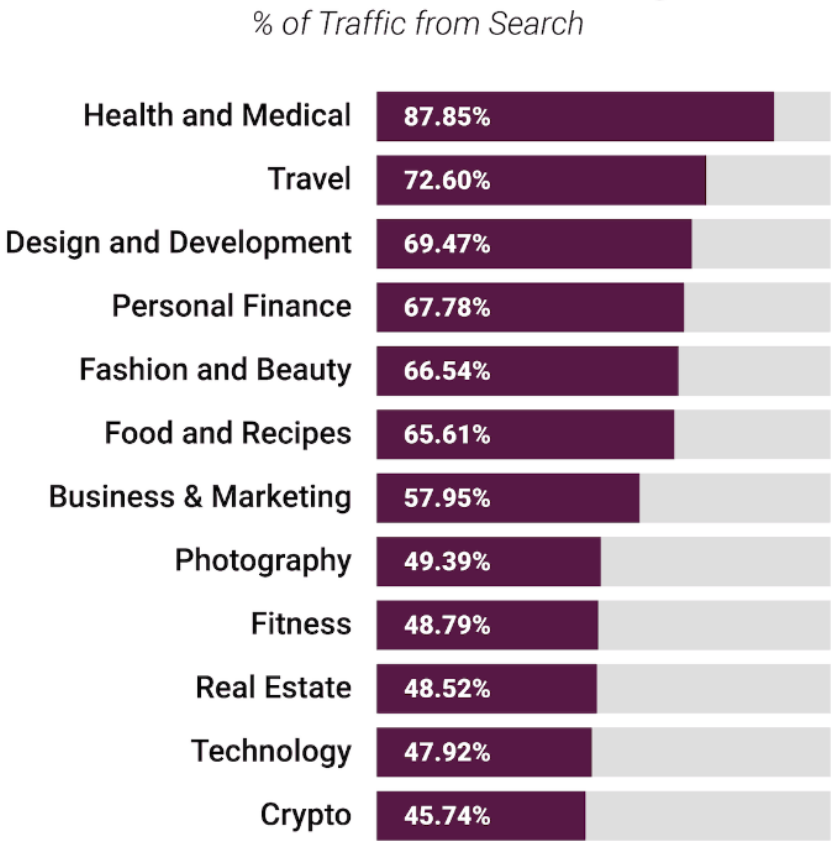
(Bradley, 2016)

- Web Search Engines (SE) are the most visible / well-known IR applications.
- They are among the most used and largest IR systems on the planet.
- Rather that return documents, search engines output an ordered list of HTML links (URLs) ranked according to their relevance.

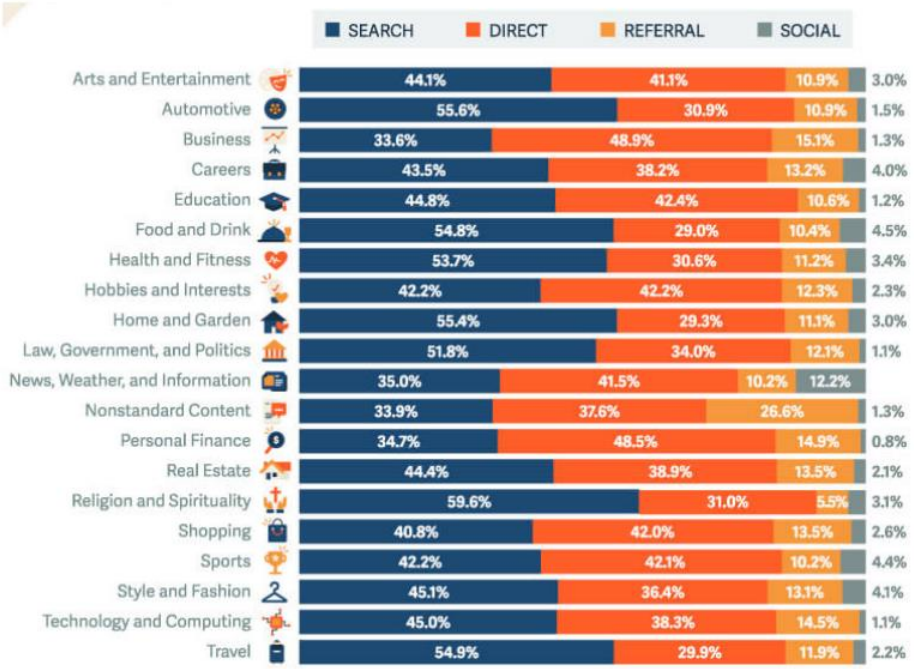
Purpose of search engines

- The web doubles in size approximately every 3 years.
- Search engines act as **FILTERS** for the vast amount of context available online.
- If relevant web content is not easily accessible it can lead to **INFORMATION OVERLOAD**.
- Search engines rank 2nd – behind email, as the most frequently used internet application.
- We study them because users of our site will most likely find us through SEs – having a knowledge of how they work is critical.

Proportion of traffic from search (organic and paid)

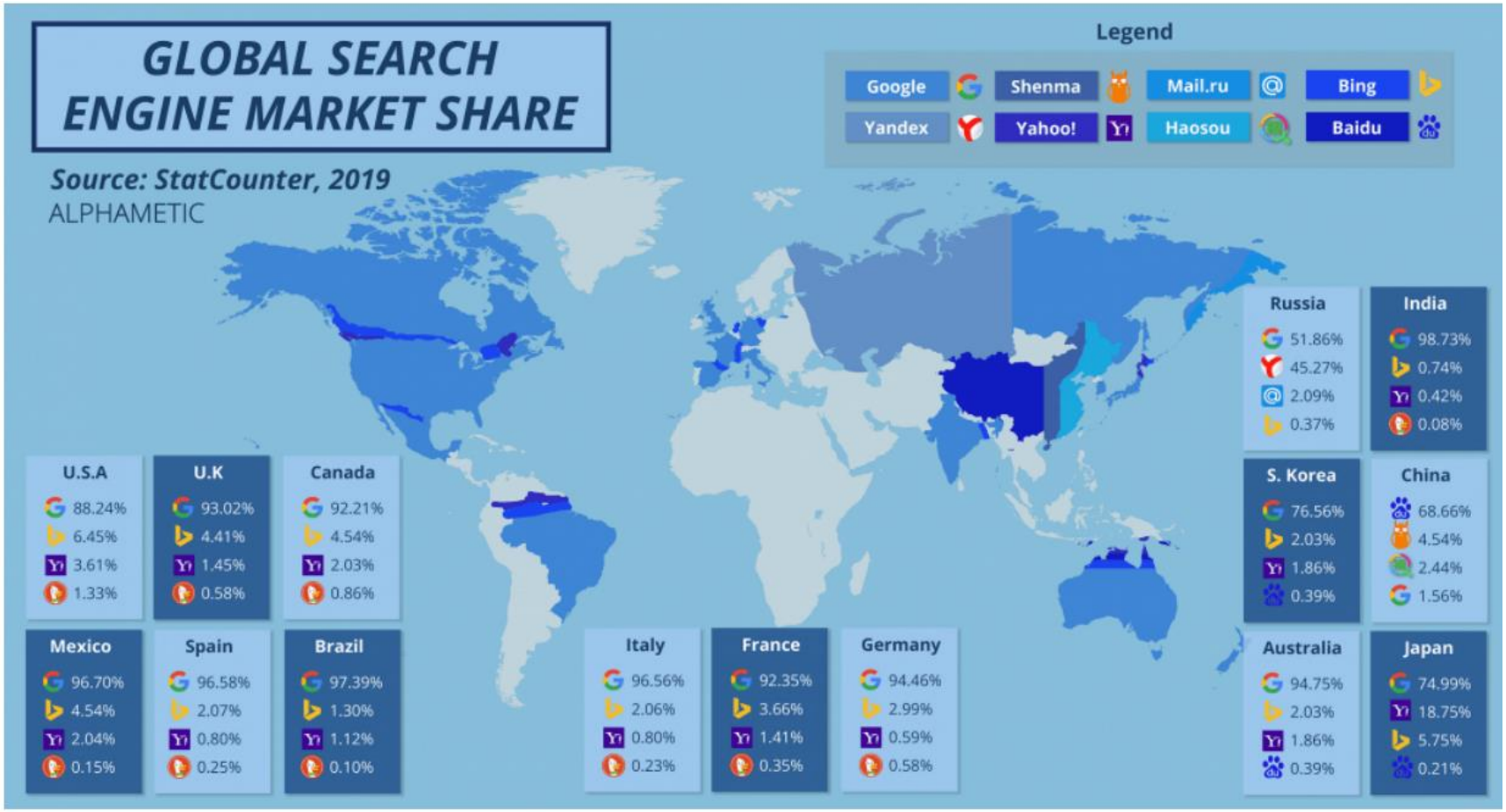


(GrowthBadger, 2020)



(Alexa, 2021)

Search engines around the world…



(Alphametic, 2020)

The search engine market

Today's search engine (SE) market is mostly dominated by a few big names although several smaller players exist.

- This can partially be explained by (1) the capital-intensive nature of providing a search engine service, the (2) integration of search with other internet products and (3) decades of IP development/refinement.

META-SEARCH ENGINES

- Some search engines, known as meta-search engines, provide their own results based on a mixture of the results of some of the others (DuckDuckGo). While originally most search engines were part of some form of web portal, nearly all now use the simplified Google "inspired" layout.

How does a SE work?

- **Which one of the following two statements do you think it is closer to truth?**

A. When a user submits a query to a SE, the SE starts searching the web to identify documents that could answer the specific query.

OR

B. When a new page / document is added on the web, (submitted to a SE) a SE analyses this document and extracts information in order to use it if a relevant query will come up.

Key web search engine components

Search engines (as information retrieval systems in their own right), share many of the same components, with a couple of key differences.

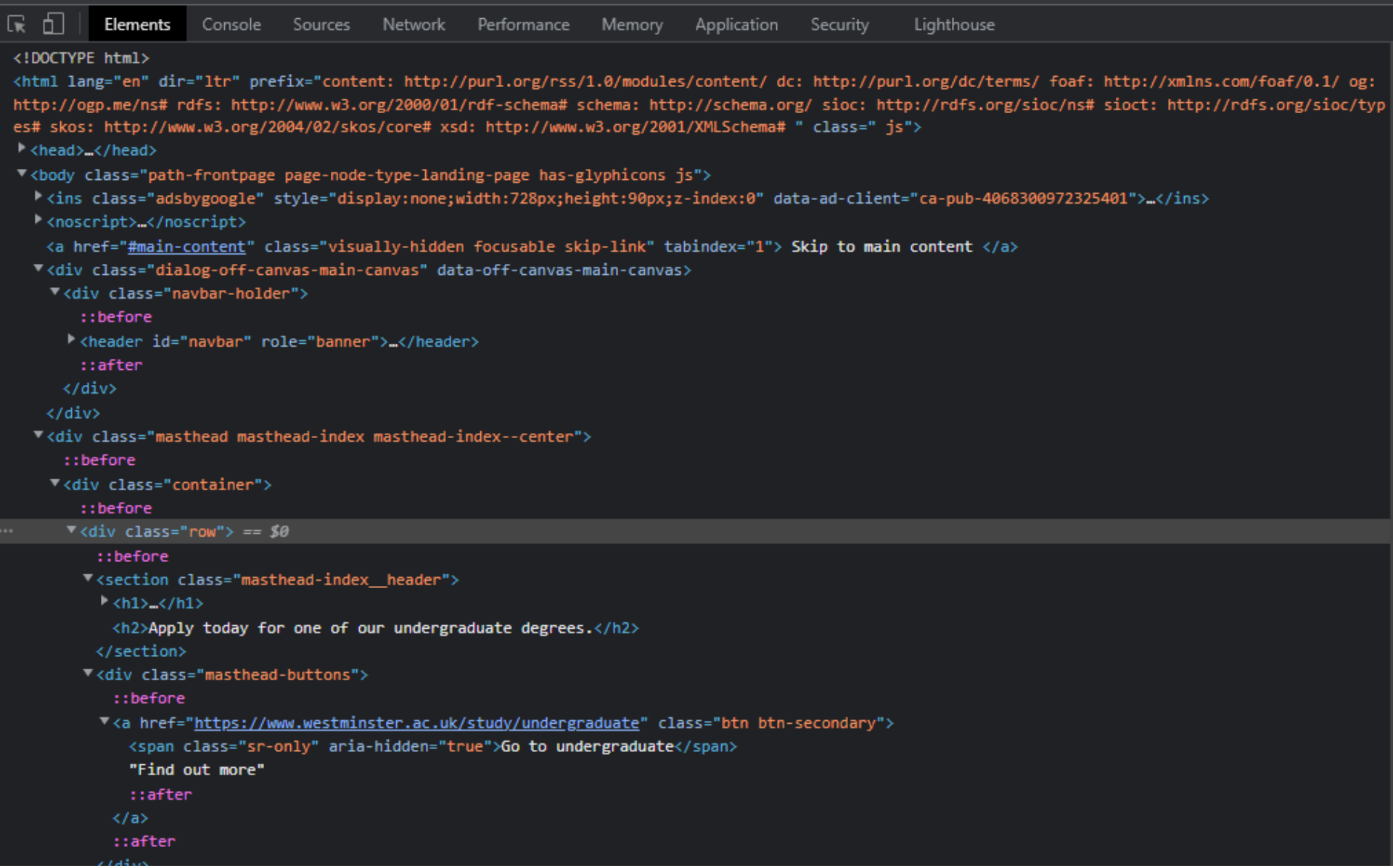
The generalised architecture of a SE includes:

- Query Input
- Query pre-processing
- A retrieval model (handling results ranking and merging)
- An Index (and supplemental Index)
- A database of web documents or fragments (for search previews)
- Web crawler (or search engine spider)

The role of the crawler

Search Engines use web crawlers (spiders) to index websites.

- When a website page is submitted to a search engine (by completing their required submission page), the search engine spider will **index** the entire site.
- A 'spider' is an automated program that is run by the search engine system. A Spider visits a web site, reads the content on the actual site, the site's Meta tags and also follows the internal links through the site.
- The spider then returns all that information back to a central repository, where the data is indexed. It will follow each link that it finds on each page and add them to the queue of sites to visit later.



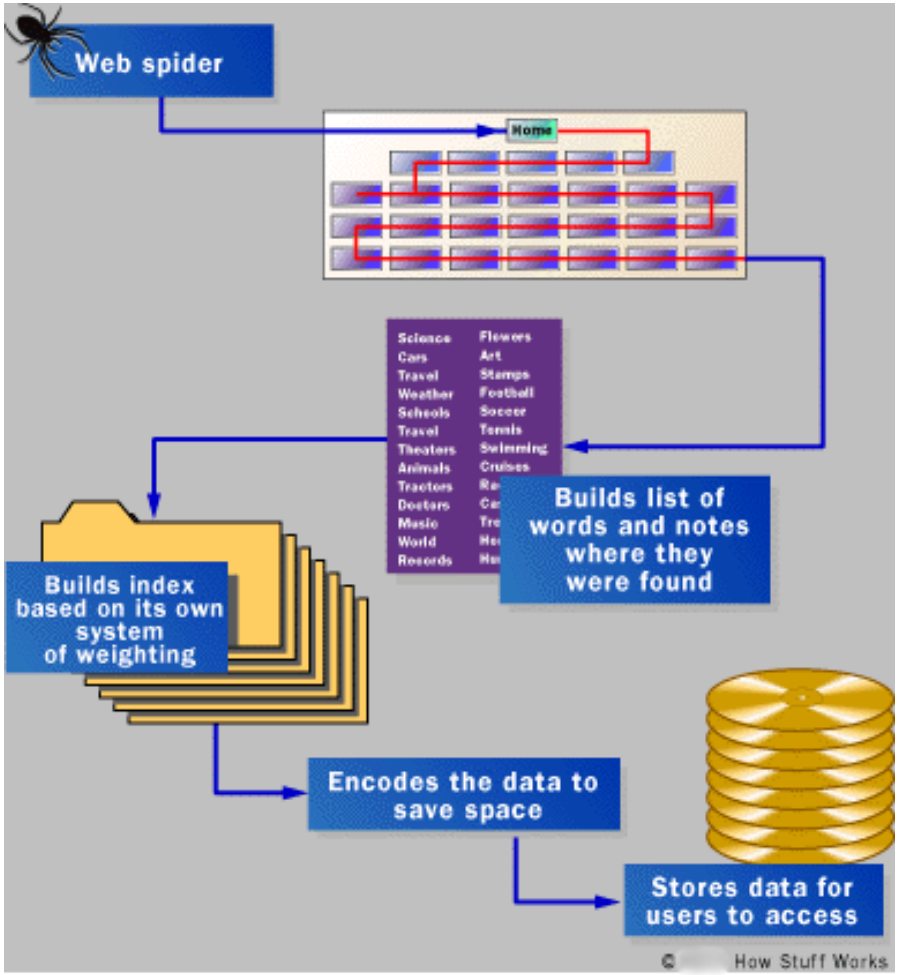
In the Chrome browser, we can press F12 to access the developer console and view the HTML code from the perspective of the “Spider”

Updating the index

- The spider will **periodically** return to the sites to check for any information that has changed.
- The frequency with which this happens is determined by the **moderators** of the search engine.

- The update frequency may also depend on:
- How old the site is. (why?)
- The niche or category of the website. (which types of websites might need updating more often?)

Graphically



from: http://computer.howstuffworks.com/internet/basics/search-engine1.htm

- In practice, search engines will distribute the INDEX and DATABASE of web pages across multiple computers using a process known as **sharding**.
- When a new QUERY is received, the QUERY is broadcast to a group of shards. Each **shard** sends back matching documents (a result set) which are then merged together and ranked.

Retrieval models

- Once pages are in the INDEX is remains to **rank and order** them when a QUERY arrives.
- To increase **performance**, some pre-ordering and ranking is performed before QUERIES are matched against the INDEX (Optimisation)
- Similarly, the INDEX might be **pruned** to increase the quality of INDEX by removing sites associated with spam, illegal activity, duplicate content, thin-content or attempted gaming of the search results.
- Like Coca-Cola, the exact formula search engines use to calculate relevance of a web page to a QUERY are **closely guarded secrets**.

Extensions to retrieval models

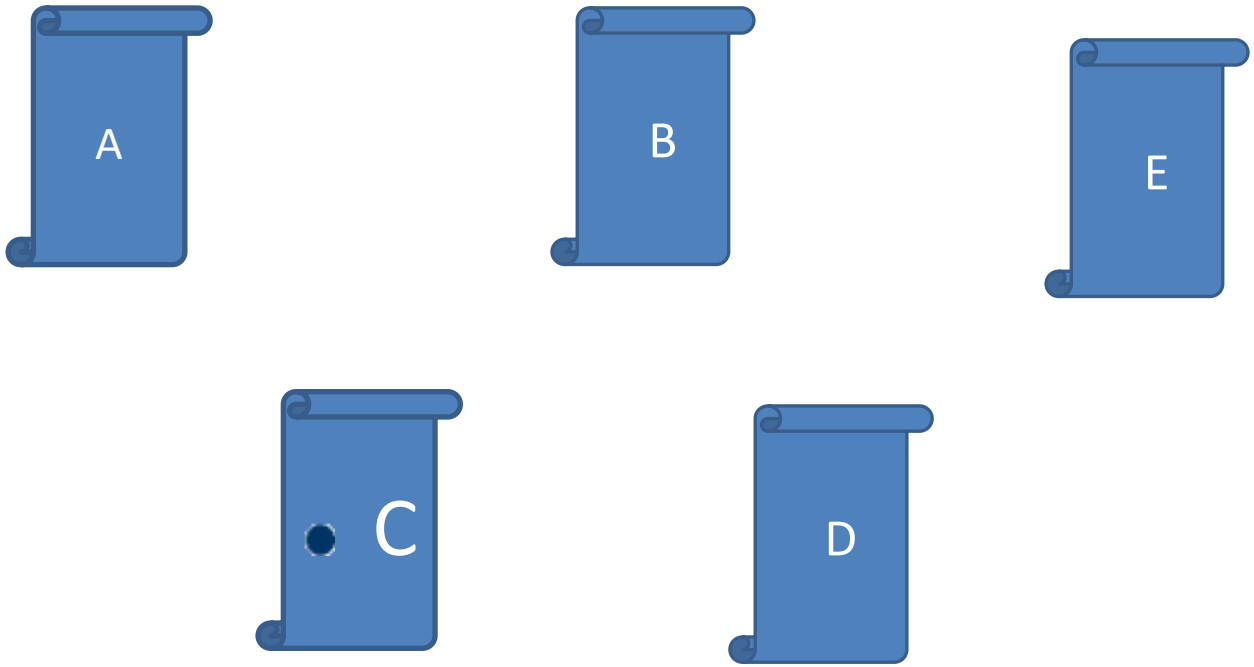
(Wired, 1998)

- One of the most important improvements to search engine retrieval models was brought about by the development of the **PageRank**™ algorithm.
- The **PageRank**™ algorithm was developed by Google Engineers (Larry Page [L]and Sergey Brin [R] and was published as an academic paper in January 1998.

PageRank – working principles

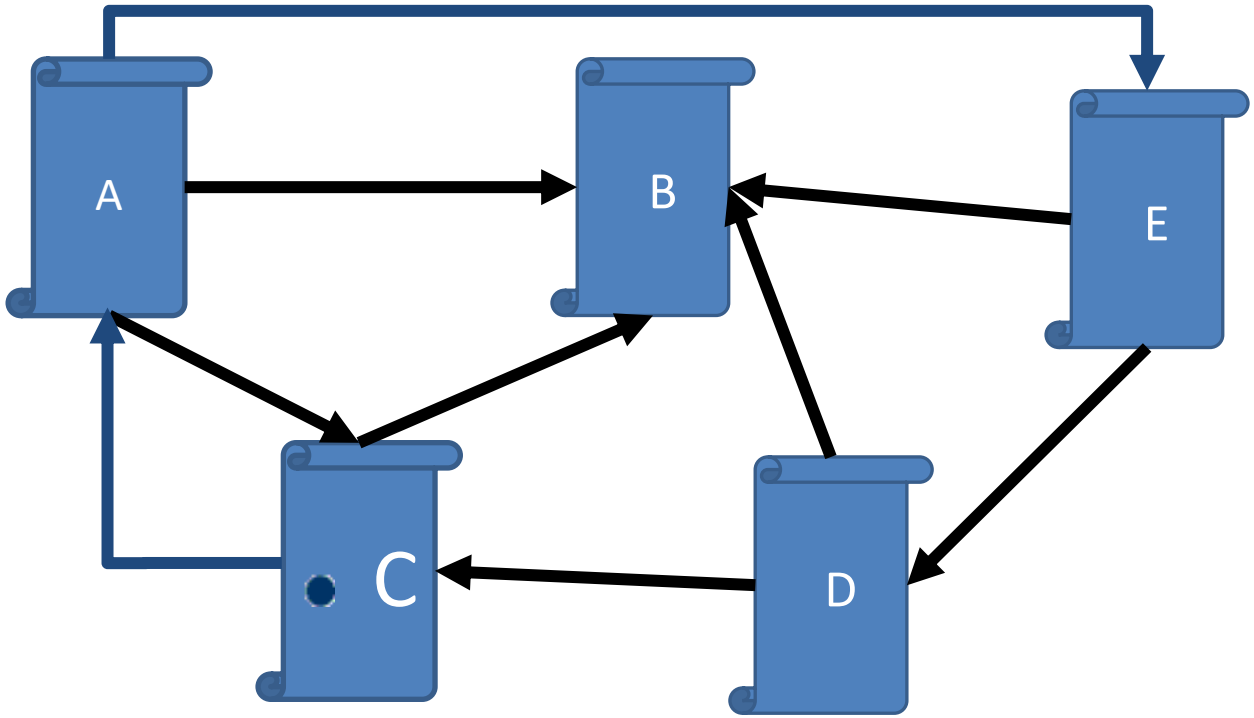
- The relevance of a web page to a search query is **SUBJECTIVE**
- It is possible to supplement the SUBJECTIVE relevance of a page with an **OBJECTIVE** measure of its quality.
- Each page on the web contains a number of **OUTBOUND** links to other pages and websites.
- Each website therefore has an arbitrary number of **INBOUND** links pointing to its home page or any one of its sub-pages.
- Since links are created manually, this implies that a website has some **confidence or expectation** that another site is relevant or of value.
- The greater the number of **INBOUND** links to a website, the greater the confidence in that website by the collective web as a whole.

PageRank – example 1



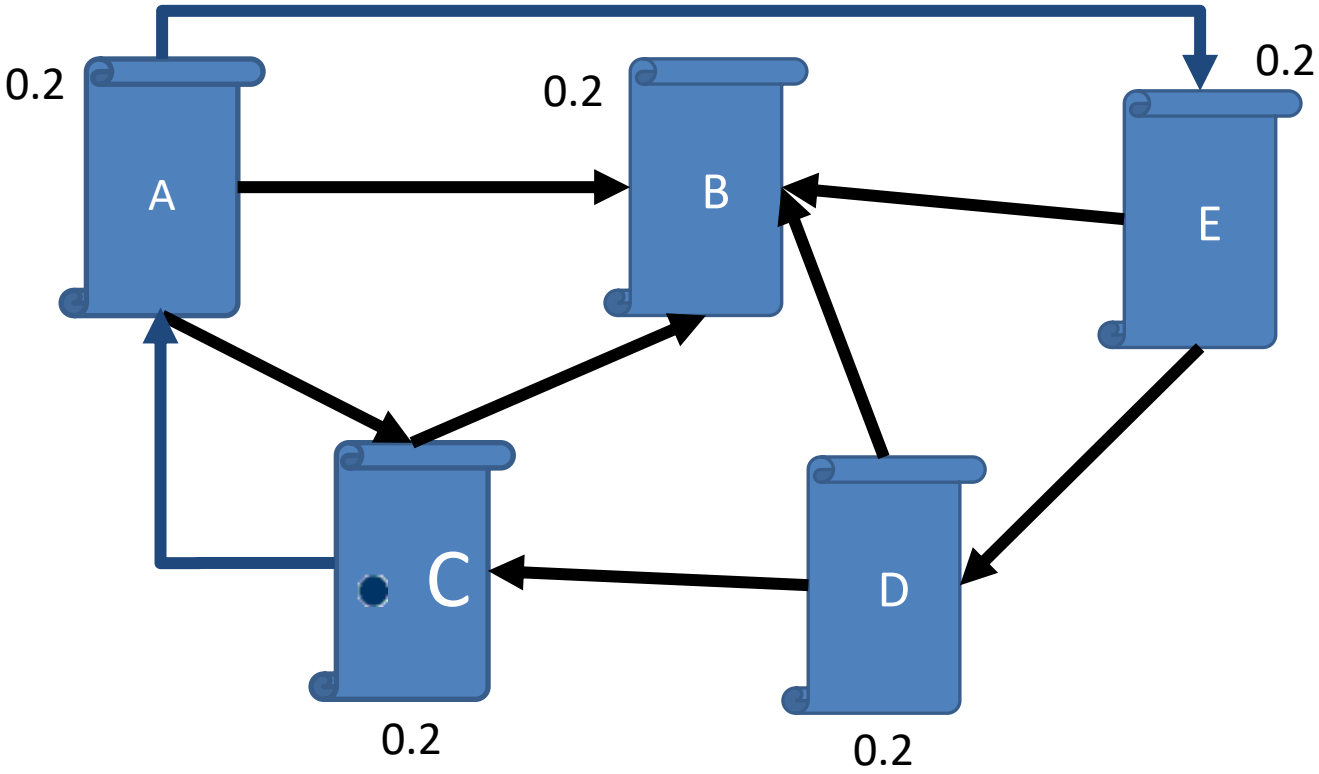
Gather together the list of pages/websites for consideration (spider)

PageRank – example 2



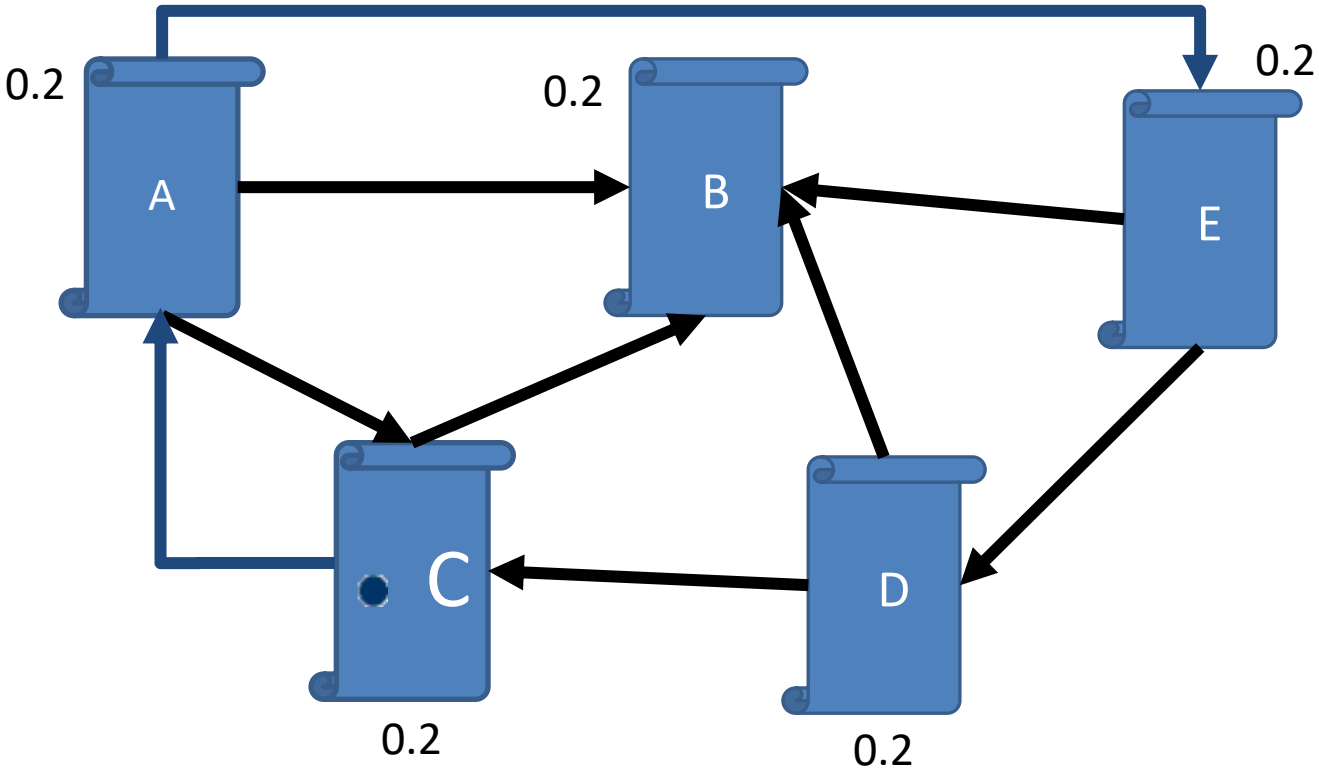
Record the links made between each website/page

PageRank – example 3



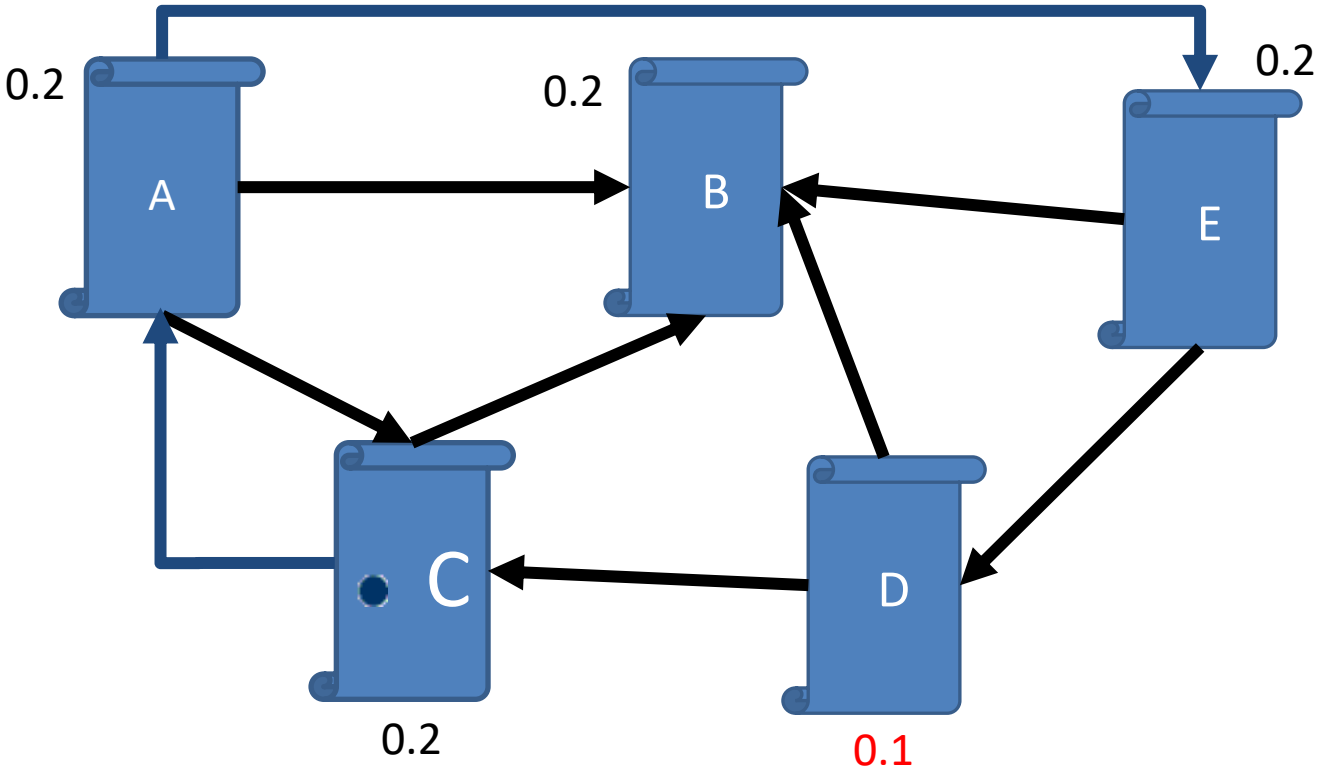
Assign an initial PageRank™ of 1/N to each website/page i.e. $\sum_i^N Pr_i = 1$

PageRank – example 4



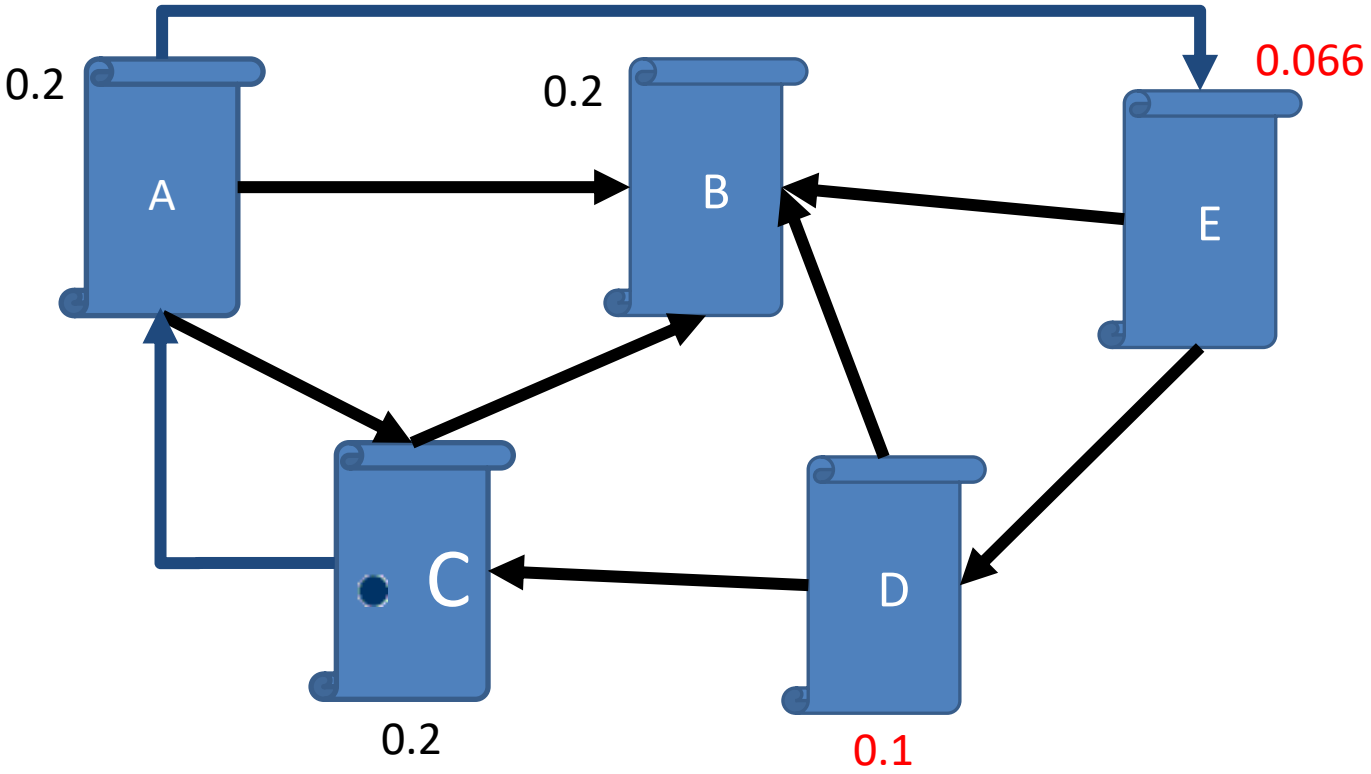
Iteratively update each website/pages PageRank™ by summing a proportion of the PageRank from any pages pointing to it.

PageRank – example 5



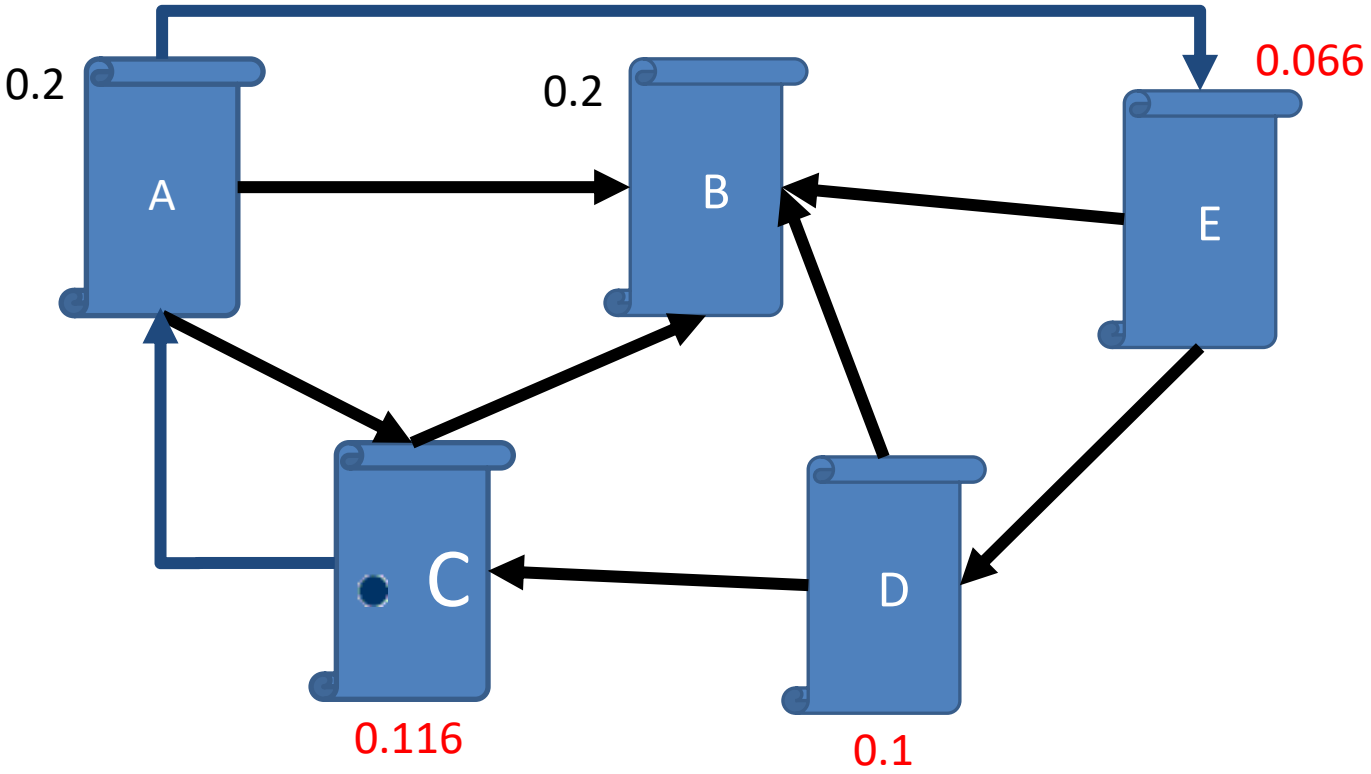
PageRank(D) = PageRank(E)/2 = 0.2 / 2 = 0.1

PageRank – example 6



PageRank(E) = PageRank(A) / 3 = 0.066

PageRank – example 7



PageRank(C) = PageRank(D) / 2 + PageRank(A) / 3 = 0.05 + 0.066

PageRank

- The **PageRank** of a page depends on the PageRank of all other pages in the network.
- After many thousands of **iterations**, the PageRank values of each page begin to stabilise close to their true values.
- The PageRank algorithm is somewhat difficult to “game” and is somewhat self-policing since it depends on links being created from either:
  - A large number of websites with a low PageRank
  - A small number of websites with a high PageRank

Semantics and language - Key extensions to retrieval models (2)

Language presents a **challenge** for search engines because it is difficult to represent the intricacies and semantics of speech in a binary system.

- ❑ Syntax (HOW we say something)
  - Colloquialisms
  - Sentence structure and order
  - Abbreviations and short hands
- ❑ Semantic (MEANING of what we say i.e. Hungry -> Food -> Take Away)
- ❑ Pragmatics and intention (**RELATIVE meaning**: The interpretation of a phrase/sentence is a largely inferential process, i.e. tax rules -> punishment for not paying tax)

Pragmatics – Little test

- **Are you albe to raed tihs sneetcne? Mdoren cmpotures fnid tihs vrey cahllneigng but stdiues hvae sohwn hmunas can esaliy do it.**
- This is because as humans we use our own knowledge and experience to interpret text.

Semantics and language - Key extensions to retrieval models (2)

A simple test of a search engine proposed by Barney Pell, previously CEO of powerset, a company that went on to develop a natural language search engine.

- Consider the **TWO** search queries:
  1. “book by children”
  2. “book about children”

Do we agree that these two search queries are fundamentally different and we would expect mutually exclusive sets of results?

- **What happens in practice?**

Semantics and language - Key extensions to retrieval models (2)

Natural Language (NL) processing is a growing field of computer science that provides tools and methodologies for working with text.

- Word stemming and lemmatisation (identify the core meaning of a word)
- Parts of Speech tagging (to identify the role of each word in a query)
- Named entity recognition (to identify people or places)
- Semantic tables and ontologies (identifying other words that have the same meaning or implication through WordNet)
- NLTK is a popular natural language toolkit for Python

In Summary

- We have reviewed some important principles of the web and the evolution of search on the web.
- Search engines provide an ability to query the web and identify relevant web pages or content.
- Search engines, as information retrieval systems themselves, borrow many of the core concepts and components of IR systems.
- Search engines are responsible for circa 40-80% of traffic to modern websites, thus they play a key role in driving acquisition.
- The search engine market in many countries is oligopolistic in nature.
- Search engines use complex retrieval models, the most famous of which involves the development of the PageRank algorithm in 1998.
- Next week we'll present and discuss the most common techniques to optimize a search engine in order to make a web site more visible.

End