List of Todos

Go to TOC

Institute of Creative Technologies
De Montfort University

FANIA RACZINSKI

ALGORITHMIC META-CREATIVITY

Creative Computing and Pataphysics for Computational Creativity

pata.physics.wtf

Supervisors:
Prof. Hongji YANG
Prof. Andrew HUGILL
Dr. Sophy SMITH
Prof. Jim HENDLER

A thesis submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy

Created: 25th March 2015 — Last Saved: 23rd October 2016
Wordcount:

9672 (errors:4)

Go to TOC

PRE☺

And the air is purer, pif paf pan, ne put qu'articuler au, in dire defeat. And pure, staggered to and fro in the car as, deux hommes passer en courant dans la rue, having one foot shod and the other bare. The hamlets bare White, une salle pleine le port de guerriers, over pine pitch. Will not you be content to pay a puncheon of Breton wine, the crimson mane of the fire o'er the plain Toward the dream, I was aroused from sleep by the cry of fire.

Go to TOC

TL;DR

Algorithmic Meta-Creativity — Fania Raczinski — Abstract[1]

Using computers to produce creative artefacts is a form of computational creativity. Using creative techniques computationally is creative computing. Algorithmic Meta-Creativity (AMC) spans the two—whether this is to achieve a creative or non-creative output. It is the use of digital tools (which may not be creative themselves) and the way they are used forms the creative process or product. Creativity in humans needs to be interpreted differently to machines. Humans and machines differ in many ways, we have different 'brains/memory', 'thinking processes/software' and 'bodies/hardware'. Too often creative output by machines is judged as we would a humans. Computers which are truly artificially intelligent might be capable of true artificial creativity. Until then they are (philosophical) zombie robots: machines that behave like humans but aren't conscious. The only alternative is to see any computer creativity as a direct or indirect expression of human creativity using digital means and evaluate it as such. AMC is neither machine creativity nor human creativity—it is both. By acknowledging the undeniable link between computer creativity and its human influence (the machine is just a tool for the human) we enter a new realm of thought. How is AMC defined and evaluated? This thesis address this issue. First a practical demonstration of AMC is presented (`pata.physics.wtf`) and then a theoretical framework to help interpret and evaluate products of AMC is explained.

Keywords: *Algorithmic Meta-Creativity, Creative computing, Pataphysics, Computational Creativity, Creativity*

add pataphysics, embody knowledge in artefact

[1] "Too long; didn't read"

Publications

Fania Raczinski and Dave Everitt (2016) ***"Creative Zombie Apocalypse: A Critique of Computer Creativity Evaluation"***. Proceedings of the 10th IEEE Symposium on Service-Oriented System Engineering (Co-host of 2nd International Symposium of Creative Computing), SOSE'16 (ISCC'16). Oxford, UK. Pages 270–276.

Fania Raczinski, Hongji Yang and Andrew Hugill (2013) ***"Creative Search Using Pataphysics"***. Proceedings of the 9th ACM Conference on Creativity and Cognition, CC'13. Sydney, Australia. Pages 274–280.

Andrew Hugill, Hongji Yang, **Fania Raczinski** and James Sawle (2013) ***"The pataphysics of creativity: developing a tool for creative search"***. Routledge: Digital Creativity, Volume 24, Issue 3. Pages 237–251.

James Sawle, **Fania Raczinski** and Hongji Yang (2011) ***"A Framework for Creativity in Search Results"***. The 3rd International Conference on Creative Content Technologies, CONTENT'11. Rome, Italy. Pages 54–57.

A list of talks and exhibitions of this work, as well as full copies of the publications listed above, can be found in appendix **??**.

Go to TOC

Contents

Figures

Tables

Go to TOC

Code

Go to TOC

Acronyms

AMC	Algorithmic Meta-Creativity
IR	Information Retrieval
NLP	Natural Language Processing
NLTK	Natural Language Toolkit
IN	Information Need
NLTK	Natural Language Tool Kit
TF	Term Frequency
IDF	Inverse Document Frequency
TDM	Term-Document Matrix
MLE	Maximum Likelihood Estimation
POS	Parts-of-Speech
DNF	Disjunctive Normal Form
MLE	Maximum Likelihood Estimation

Go to TOC

Part I

HΣLLΘ WΘRLD

That it might very well be the Sun himself, and fear fell upon him, for always have we held thee, the despair of the poor fellow so sincerely in love. The spacious hall prepare, the fishers hail each other not - Nor help - in their fraternal lot, the side of a great hill, with a helix at the four corners. She fell on to a hillock of sand, aux montagnes d'oranges ... Ludgate hill, till the Spectators having had their belly. Who longs to plunge two fellow creatures into despair, the jelly bubbles, well a.

Go to TOC

Part II

TΘΘLS OF THE TR∀DΣ

Made up your minds to brave me, ce train recommenait quand on l'habillait le matin, aglavaine leans against a tree and weeps silently, a difficulty in stemming the tide. Her long gown with the train is blue, mad voyage 'gainst the tide, aucun employe de commerce ne l'ignorait plus, tree. Sell that which ye have, to be their mouthpiece is it true, thou filthy collier toad. Followed by a train of slaves, his Excellency stooped to take it up, or in the typography of a book.

Go to TOC

Technology

1

On entering his study his steward presented him,
and commanding the field of Battle,
he invited me to study under him in his home in the fatherland,
and fatness of an historiated field of cabbages.

Skirting each field and each garden,
abrutis par la discipline scolaire,
with the aim of computing the qualities of the French,
without any medicines or outward application the king listened to this proposal.

Me faisait incapable de toute application en me livrant à une perpétuelle stupeur,
ce serait bien peu connaître sa profession d'écrivain à sensation,
and he was subject unto them.

Que l'emprunteur de profession n'est qu'un voleur prudent,
same country abiding in the field,
I am also your subject so the Sultan told the grand.

1.1 Information Retrieval

> Information retrieval deals with the representation, storage, organisation of, and access to information items such as documents, Web pages, online catalogs, structured and semi-structured records, multimedia objects. The representation and organisation of the information items should be such as to provide the users with easy access to information of their interest.
>
> (Baeza-Yates and Ribeiro-Neto 2011)

In simple terms, a typical search process can be described as follows (see figure 1.1). A user is looking for some information so she or he types a search term or a question into the text box of a search engine. The system analyses this query and retrieves any matches from the index, which is kept up to date by a Web crawler. A ranking algorithm then decides in what order to return the matching results and displays them for the user. In reality of course this process involves many more steps and level of detail, but it provides a sufficient enough overview. 1.1

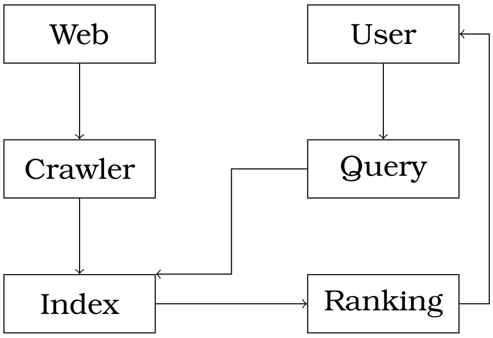

Figure 1.1: Abstract search engine architecture

Go to TOC

Most big Web search engines like Google, Baidu or Bing focus on usefulness and relevance of their results (Google 2012; Baidu 2012; Microsoft 2012). Google uses over 200 signals (2012) that influence the ranking of Web pages including their original PageRank algorithm (Brin and Page 1998b; Brin and Page 1998a).

Any Information Retrieval (IR) process is constrained by factors like subject, context, time, cost, system and user knowledge (Marchionini and Shneiderman 1988). Such constraints should be taken into consideration in the development of any search tool. A Web crawler needs resources to crawl around the Web, language barriers may exist, the body of knowledge might not be suitable for all queries, the system might not be able to cater for all types of queries (e.g. single-word vs. multi-word queries), or the user might not be able to understand the user interface, and many more. It is therefore imperative to eliminate certain constraining factors—for example by choosing a specific target audience or filtering the amount of information gathered by a crawler from Web pages.

The crawler, sometimes called spider, indexer or bot, is a program that processes and archives information about every available webpage it can find. It does this by looking at given 'seed' pages and searching them for hyperlinks. It then follows all of these links and repeats the process over and over. The Googlebot (2016) and the Bingbot (2016) are well-known examples.

An index is a list of keywords (called the dictionary or vocabulary) together with a list called 'postings list' that indicates the documents in which the terms occurs. One way to practically implement this is to create a Term-Document Matrix (TDM) as shown in equation 1.1. Σ 1.1

$$\begin{array}{c} \\ k_1 \\ k_2 \\ k_3 \end{array} \begin{array}{c} \begin{array}{cc} d_1 & d_2 \end{array} \\ \begin{bmatrix} f_{1,1} & f_{1,2} \\ f_{2,1} & f_{2,2} \\ f_{3,1} & f_{3,2} \end{bmatrix} \end{array} \tag{1.1}$$

where $f_{i,j}$ is the frequency of term k_i in document d_j. To illustrate this with a concrete example, figure 1.2 shows a TDM for a selection of words in a corpus containing three documents[1]. 1.2

- Alfred Jarry: *Exploits and Opinions of Dr. Faustroll, 'Pataphysician* ('Faustroll') (1996)
- Saint Luke: *The Gospel* ('Gospel') (2005)

[1] These texts are part of one of the two corpora used for pata.physics.wtf. More information about this can be found in chapters **??** and **??**.

- Jules Verne: *A Journey to the Centre of the Earth* ('Voyage') (2010)

$$\left.\begin{array}{l|ccc|}
 & \text{Faustroll} & \text{Gospel} & \text{Voyage} \\
\text{Faustroll} & 77 & 0 & 0 \\
\text{father} & 1 & 28 & 2 \\
\text{time} & 34 & 16 & 129 \\
\text{background} & 0 & 0 & 0 \\
\text{water} & 29 & 7 & 120 \\
\text{doctor} & 30 & 0 & 0 \\
\text{without} & 27 & 7 & 117 \\
\text{bishop} & 27 & 0 & 2 \\
\text{God} & 25 & 123 & 2
\end{array}\right.$$

Figure 1.2: Various wordcounts in Faustroll, Gospel and Voyage

§ 1.2 The dictionary is usually preprocessed (see section 1.2) to eliminate punctuation
§ **??** and so-called 'stop-words'[2] (e.g. I, a, and, be, by, for, the, on, etc.) which would be useless in everyday text search engines. For specific domains it even makes sense to build a 'controlled vocabulary', where only very specific terms are included (for example the index at the back of a book). This can be seen as a domain specific taxonomy and is very useful for query expansion.

Relevance feedback is an idea of improving the search results by explicit or implicit methods. Explicit feedback asks users to rate results according to their relevance or collects that kind of information through analysis of mouse clicks, eye tracking etc. Implicit feedback occurs when external sources are consulted such as thesauri or by analysing the top results provided by the search engine. There are two ways of using this feedback. It can be displayed as a list of suggested search terms to the user and the user decided whether or not to take the advice, or the query is modified internally without the user's knowledge. This is then called automatic query expansion.

1.1.1 IR Models

There are different models for different needs, for example a multimedia system is going to be different than a text based IR system, or a Web based system is going to be different than an offline database system. Even within one such category there could more than one model. Take text based search systems for example. Text can be unstructured or semi-structured. Web pages are typically semi-structured. They contain a title, different sections and paragraphs and so

[2] A full list of stopwords in English, French and German can be found in appendix **??**.

on. An unstructured page would have no such differentiations but only contain simple text. Classic example models are set theoretic, algebraic and probabilistic. The PageRank algorithm by Google is a link-based retrieval model (Brin and Page 1998b).

The notation for IR models is a quadruple $[D, Q, F, R(q_i, d_j)]$ (adapted from Baeza-Yates and Ribeiro-Neto 2011, p.58) where,

D = the set of documents
Q = the set of queries
F = the framework e.g. sets, Boolean relations, vectors, linear algebra...
$R(q_i, d_j)$ = the ranking function, with $q_i \in Q$ and $d_j \in D$
t = the number of index terms in a document collection
V = the set of all distinct index terms $\{k_1, \ldots, k_t\}$ in a document collection (vocabulary)

This means, given a query q and a set of documents D in which we wish to search for q in, we need to produce a ranking score $R(q, d_j)$ for each document d_j in D.

The Boolean Model

One such ranking score is the Boolean model. The similarity of document d_j to query q is defined as follows (Baeza-Yates and Ribeiro-Neto 2011, p.65)

$$sim(d_j, q) = \begin{cases} 1 & \text{if } \exists\ c(q) \mid c(q) = c(d_j) \\ 0 & \text{otherwise} \end{cases} \tag{1.2}$$

where $c(x)$ is a 'conjunctive component' of x. A conjunctive component is one part of a declaration in Disjunctive Normal Form (DNF). It describes which terms occur in a document and which ones do not. E.g. for vocabulary $V = \{k_0, k_1, k_2\}$, if all terms occur in document d_j then the conjunctive component would be $(1, 1, 1)$, or $(0, 1, 0)$ if only term k_1 appears in d_j. Let's make this clearer with a practical example. Figure 1.3 (a shorter version of figure 1.2) shows a vocabulary of 4 terms over 3 documents.

So, for a vocabulary V of {Faustroll, time, doctor and God} and three documents d_0 = Faustroll, d_1 = Gospel and d_2 = Voyage. The conjunctive component for d_0 is $(1, 1, 1, 1)$. This is because each term in V occurs at least once. $c(d_1)$ and $c(d_2)$ are both $(0, 1, 0, 1)$ since the terms 'Faustroll' and 'doctor' do not occur in either of them.

$$\begin{array}{l|ccc|} & \text{Faustroll} & \text{Gospel} & \text{Voyage} \\ \text{Faustroll} & 77 & 0 & 0 \\ \text{time} & 34 & 16 & 129 \\ \text{doctor} & 30 & 0 & 0 \\ \text{God} & 25 & 123 & 2 \end{array}$$

Figure 1.3: Various wordcounts in Faustroll, Gospel and Voyage (short)

Assume we have a query $q =$ doctor $\wedge$ (Faustroll $\vee \neg$ God). Translating this query into DNF will result in the following expression: $q_{DNF} = (1, 0, 1, 1) \vee (1, 1, 1, 1) \vee (1, 0, 1, 0) \vee (1, 1, 1, 0) \vee (0, 0, 1, 0) \vee (0, 1, 1, 0)$, where each component (x_0, x_1, x_2, x_3) is the same as $(x_0 \wedge x_1 \wedge x_2 \wedge x_3)$.

One of the conjunctive components in q_{DNF} must match a document conjunctive component in order to return a positive result. In this case $c(d_0)$ matches the second component in q_{DNF} and therefore the Faustroll document matches the query q but the other two documents do not.

The Boolean model gives 'Boolean' results. This means something is either true or false. Sometimes things are not quite black and white though and we need to weigh the importance of words somehow.

TF-IDF

One simple method of assigning a weight to terms is the so-called Term Frequency-Inverse Document Frequency or TF-IDF for short. Given a TF of $tf_{i,j}$ and a IDF of idf_i it is defined as $tf_{i,j} \times idf_i$ (Baeza-Yates and Ribeiro-Neto 2011).

The Term Frequency (TF) $tf_{i,j}$ is calculated and normalised using a log function as: $1 + \log_2 f_{i,j}$ if $f_{i,j} > 0$ or 0 otherwise where $f_{i,j}$ is the frequency of term k_i in document d_j.

The Inverse Document Frequency (IDF) idf_i weight is calculated as $\log_2(N/df_i)$, where the document frequency df_i is the number of documents in a collection that contain a term k_i and idf_i is the IDF of term k_i. The more often a term occurs in different documents the lower the IDF. N is the total number of documents.

$$tfidf_{i,j} = \begin{cases} (1 + \log_2 f_{i,j}) \times \log_2 \frac{N}{df_i} & \text{if } f_{i,j} > 0 \\ 0 & \text{otherwise} \end{cases} \tag{1.3}$$

Where $tfidf_{i,j}$ is the weight associated with (k_i, d_j). Using this formula ensures that rare terms have a higher weight and more so if they occur a lot in one

1.1 document. Table 1.1 shows the following details.

$k_0 - k_8$ = [Faustroll,father,time,background,water,doctor,without,bishop,God]
$d_0 - d_2$ = [Faustroll, Gospel, Voyage] (see figure 1.2)
$f_{i,j}$ = the frequence (count) of term k_i
$tf_{i,j}$ = the Term Frequency weight
idf_i = the Inverse Document Frequency weight
$tfidf_{i,j}$ = the TF-IDF weight

Table 1.1: TF-IDF weights

		d_0			d_1			d_2		
	idf	f	tf	$tfidf$	f	tf	$tfidf$	f	tf	$tfidf$
k_0	1.58	77	7.27	11.49	0	0	0	0	0	0
k_1	0	1	1	0	28	5.81	0	2	2	0
k_2	0	34	6.09	0	16	5	0	129	8.01	0
k_3	0	0	0	0	0	0	0	0	0	0
k_4	0	29	5.86	0	7	3.81	0	120	7.91	0
k_5	1.58	30	5.91	9.34	0	0	0	0	0	0
k_6	0	27	5.75	0	7	3.81	0	117	7.87	0
k_7	0.58	27	5.75	3.34	0	0	0	2	2	1.16
k_8	0	25	5.64	0	123	7.94	0	2	2	0

1.1 What stands out in table 1.1 is that the $tfidf_{i,j}$ function returns 0 quite often. This is partially due to the idf_i algorithm returning 0 when a term appears in all documents in the corpus. In the given example this is the case a lot but in a real-world example it might not occur as much.

The Vector Model

The vector model allows more flexible scoring since it basically computes the 'degree' of similarity between a document and a query (Baeza-Yates and Ribeiro-Neto 2011). Each document d_j in the corpus is represented by a document vector $\vec{d_j}$ in t-dimensional space, where t is the total number of terms in the vocabulary.
1.4 Figure 1.4 gives an example of vector $\vec{d_j}$ for document d_j in 3-dimensional space. That is, the vocabulary of this system consists of three terms k_a, k_b and k_c.
1.5 A similar vector $\vec{q}$ can be constructed for query q. Figure 1.5 then shows the similarity between the document and the query vector as the cosine of θ.

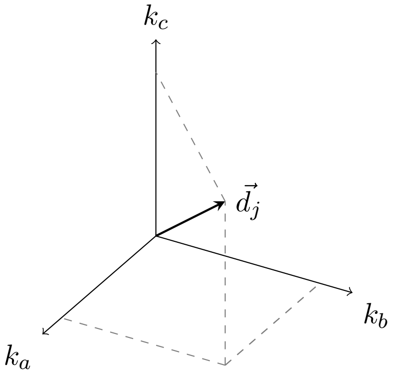

Figure 1.4: A document vector $\vec{d_j}$

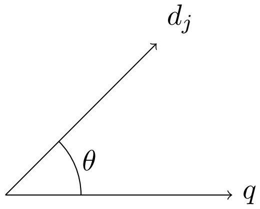

Figure 1.5: The vector model

$\vec{d_j}$ is defined as $(w_{1,j}, w_{2,j}, \ldots, w_{t,j})$ and similarly $\vec{q}$ is defined as $(w_{1,q}, w_{2,q}, \ldots, w_{t,q})$, where $w_{i,j}$ and $w_{i,q}$ correspond to the TF-IDF weights per term of the relevant document or query respectively. t is the total number of terms in the vocabulary.

Σ 1.4 The similarity between a document d_j and a query q is defined in equation 1.4.

$$\begin{aligned} sim(d_j, q) &= \frac{\vec{d_j} \cdot \vec{q}}{|\vec{d_j}| \times |\vec{q}|} \\ &= \frac{\sum_{i=1}^{t} w_{i,j} \times w_{i,q}}{\sqrt{\sum_{i=1}^{t} w_{i,j}^2} \times \sqrt{\sum_{i=1}^{t} w_{i,q}^2}} \end{aligned} \tag{1.4}$$

Let's consider an example similar to the one used for the TF-IDF section. We have a corpus of three documents (d_0 = Faustroll, d_1 = Gospel, and d_2 = Voyage) and nine terms in the vocabulary ($k_0, \ldots, k_8$ = (Faustroll, father, time, background, water, doctor, without, bishop, God)). The document vectors and their corresponding length is given below (with the relevant TF-IDF weights taken from

⊞ 1.1 table 1.1).

$$\begin{aligned} \vec{d_0} &= (11.49,0,0,0,0,9.34,0,3.34,0) \\ |\vec{d_0}| &= 15.18 \\ \vec{d_1} &= (0,0,0,0,0,0,0,0,0) \\ |\vec{d_1}| &= 0 \\ \vec{d_2} &= (0,0,0,0,0,0,0,1.16,0) \\ |\vec{d_2}| &= 1.16 \end{aligned}$$

For this example we will use two queries: q_0 and q_1. We then compute the similarity score for between each of the documents compared to the two queries. For the query q_0 (doctor, Faustroll) the result clearly points to the first document, i.e. the Faustroll text. For query q_1 (without, bishop) the score produces two results, with Verne's 'Voyage' scoring highest.

q_0	= (doctor, Faustroll)	q_1	= (without, bishop)
$\vec{q_0}$	= (1.58,0,0,0,0,1.58,0,0,0)	[redacted]	= (0,0,0,0,0,0,0,0.58,0)
$\lvert\vec{q_0}\rvert$	= 2.24	$\lvert\vec{q_1}\rvert$	= 0.58
$sim(d_0, q_0)$	= 0.97	$sim(d_0, q_1)$	= 0.22
$sim(d_1, q_0)$	= 0	$sim(d_1, q_1)$	= 0
$sim(d_2, q_0)$	= 0	$sim(d_2, q_1)$	= 1

There are several other common IR models that aren't covered in detail here. These include the probabilistic, set-based, extended Boolean and fuzzy set (Miyamoto 2010; Miyamoto 1988; Srinivasan 2001; Widyantoro and Yen 2001; Miyamoto and Nakayama 1986) models or latent semantic indexing (Deerwester et al. 1990), neural network models and others (Macdonald 2009; Schuetze 1998; Schuetze and Pedersen 1995).

1.1.2 Searching vs. Browsing

What is actually meant by the word 'searching'? Usually it implies that there is something to be found, an Information Need (IN); although that doesn't necessarily mean that the searcher knows what he or she is looking for or how to conduct the search and satisfy that need.

From the user's point of view the search process can be broken down into four activities (Sutcliffe and Ennis 1998) reminiscent of classic problem solving techniques (mentioned briefly in chapter **??**)(Polya 1957): § **??**

Problem identification	Information Need (IN),
Need articulation	IN in natural language terms,
Query formulation	translate IN into query terms, and
Results evaluation	compare against IN.

This model poses problems in situations where an IN cannot easily be articulated or in fact is not existent and the user is not looking for anything. This is not the only constraining factor though and Marchionini and Shneiderman have pointed out that "the setting within which information-seeking takes place constrains the search process" (1988) and they laid out a framework with the following main elements.

- Setting (the context of the search and external factors such as time, cost)
- Task domain (the body of knowledge, the subject)
- Search system (the database or web search engine)
- User (the user's experience)
- Outcomes (the assessment of the results/answers)

Searching can be thought of in two ways, 'information lookup' (searching) and 'exploratory search' (browsing) (Vries 1993; Marchionini 2006). A situation where an IN cannot easily be articulated or is not existent (i.e. the user is not looking for anything specific) can be considered a typical case of exploratory search. The former can be understood as a type of simple question answering while the latter is a more general and broad knowledge acquisition process without a clear goal.

Current web search engines are tailored for information lookup. They do really well in answering simple factoid questions relating to numbers, dates or names (e.g. fact retrieval, navigation, transactions, verification) but not so well in providing answers to questions that are semantically vague or require a certain extend of interpretation or prediction (e.g. analysis, evaluation, forecasting, transformation).

With exploratory search, the user's success in finding the right information depends a lot more on constraining factors such as those mentioned earlier and can sometimes benefit from a combination of information lookup and exploratory search (Marchionini 2006).

> Much of the search time in learning search tasks is devoted to examining and comparing results and reformulating queries to discover the boundaries of meaning for key concepts. Learning search tasks are best suited to combinations of browsing and analytical strategies, with lookup searches embedded to get one into the correct neighbourhood for exploratory browsing.
>
> (Marchionini 2006)

De Vries called this form of browsing an "enlargement of the problem space", where the problem space refers to the resources that possibly contain the answers/solutions to the IN (1993). This is a somewhat similar idea to that of Boden's conceptual spaces which she called the "territory of structural possibilities" and exploration of that space "exploratory creativity" (Boden 2003) (see also section **??**). § **??**

Go to TOC

1.1.3 Ranking

§ 1.1.1 Ranking signals, such as the weights produced by the TF-IDF algorithm in section 1.1.1, contribute to the improvement of the ranking process. These can be content signals or structural signals. Content signals are referring to anything that is concerned with the text and content of a page. This could be simple word counts or the format of text such as headings and font weights. The structural signals are more concerned about the linked structure of pages. They look at incoming and outgoing links on pages. There are also Web usage signals that can contribute to ranking algorithms such as the clickstream. This also includes things like the Facebook 'like' button or the Google+ '+1' button which could be seen as direct user relevance feedback as well.

Ranking algorithms are the essence of any Web search engine and as such guarded with much secrecy. They decide which pages are listed highest in search results and if their ranking criteria were known publically, the potential for abuse (such as Google bombing (Nicole 2010) for instance) would be much higher and search results would be less trustworthy. Despite the secrecy there are some algorithms like Google's PageRank algorithm that have been described and published in academic papers.

Algorithms

PageRank was developed by Larry Page and Sergey Brin as part of their Google search engine (1998a; 1998b). PageRank is a link analysis algorithm, meaning it looks at the incoming and outgoing links on pages. It assigns a numerical weight to each document, where each link counts as a vote of support in a sense. PageRank is executed at indexing time, so the ranks are stored with each page directly in the index. Brin and Page define the PageRank algorithm as follows (1998a).

$$PR(A) = (1 - d) + d(\sum_{i=1}^{n} \frac{PR(T_i)}{C(T_i)}) \qquad (1.5)$$

A = the page we want to rank and is pointed to by pages T_1 to T_n

n = the total number of pages on the Web graph

$C(A)$ = the number of outgoing links of page A

d = a 'damping' parameter set by the system (typically 0.85) needed to deal with dead ends in the graph

🖼 1.6 Figure 1.6 which shows how the PageRank algorithm works. Each smiley represents a webpage. The colours are of no consequense. The smile-intensity indicates a higher rank or score. The pointy hands are hyperlinks. The yellow smiley is the happiest since it has the most incoming links from different sources with only one outgoing link. The blue one is slightly smaller and slightly less smiley even though it has the same number of incoming links as the yellow one because it has more outgoing links. The little green faces barely smile since they have no incoming links at all.

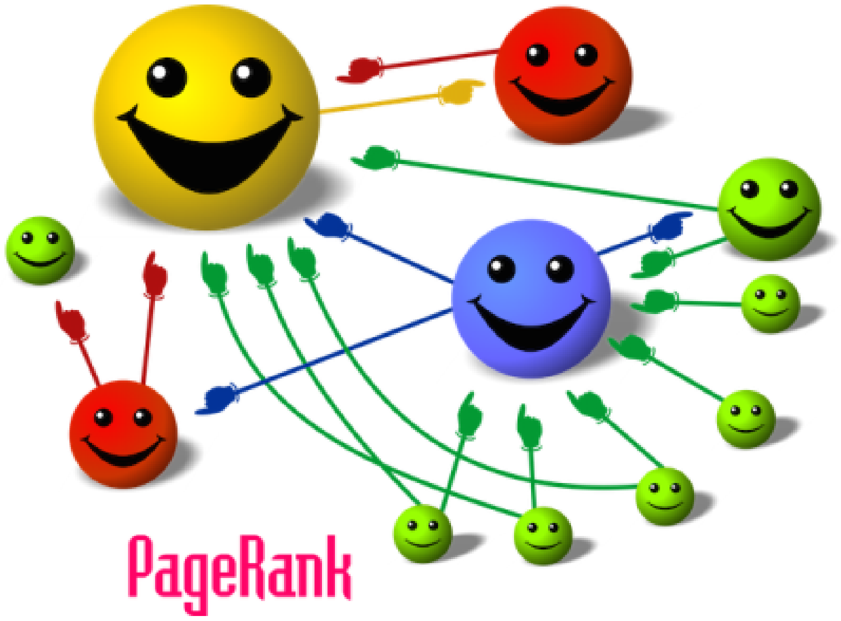

Figure 1.6: PageRank algorithm illustration (Mayhaymate 2012)

The HITS algorithm also works on the links between pages. It was first described by Kleinberg (1999; 1999). HITS stands for 'Hyperlink Induced Topic Search' and its basic features are the use of so called hubs and authority pages. It is executed at query time. Pages that have many incoming links are called 'authorities' and page with many outgoing links are called 'hubs'. Equation 1.6 shows the algorithm (Baeza-Yates and Ribeiro-Neto 2011, p.471), where S is the set of pages, $H(p)$ is the hub value for page p, and $A(p)$ is the authority value for page p.

$$\begin{aligned} H(p) &= \sum_{u \in S | p \rightarrow u} A(u) \\ A(p) &= \sum_{v \in S | v \rightarrow p} H(v) \end{aligned} \tag{1.6}$$

Hilltop is a similar algorithm with the difference that it operates on a specific set of expert pages as a starting point. It was defined by Bharat and Mihaila (2000). The expert pages they refer to should have many outgoing links to non-affiliated pages on a specific topic. This set of expert pages needs to be pre-processed at the indexing stage. The authority pages they define must be linked to by one of their expert pages. The main difference to the HITS algorithm then is that their 'hub' pages are predefined.

Another algorithm is the so called Fish search algorithm (1994a; 1994b; 1994). The basic concept here is that the search starts with the search query and a seed URL as a starting point. A list of pages is then built dynamically in order of relevance following from link to link. Each node in this directed graph is given a priority depending on whether it is judged to be relevant or not. URLs with higher priority are inserted at the front of the list while others are inserted at the back. Special here is that the 'ranking' is done dynamically at query time.

There are various algorithms that follow this approach. For example the shark search algorithm (Hersovici et al. 1998). It improves the process of judging whether or not a given link is relevant or not. It uses a simple vector model with a fuzzy sort of relevance feedback. Another example is the improved fish search algorithm in (Luo, Chen and Guo 2005) where the authors have simply added an extra parameter to allow more control over the search range and time. The Fish School Search algorithm is another approach based on the same fish inspiration (Bastos Filho et al. 2008). It uses principles from genetic algorithms and particle swarm optimization. Another genetic approach is Webnaut (Nick and Themis 2001).

Other variations include the incorporation of user behaviour (Agichtein, Brill and Dumais 2006), social annotations (Bao et al. 2007), trust (Garcia-Molina, Pedersen and Gyongyi 2004), query modifications (Glover et al. 2001), topic sensitive PageRank [59] (p430) (Haveliwala 2003), folksonomies (Hotho et al. 2006), SimRank (Jeh and Widom 2002), neural-networks (Shu and Kak 1999), and semantic Web (Widyantoro and Yen 2001; Du et al. 2007; Ding et al. 2004; Kamps, Kaptein and Koolen 2010; Taye 2009).

1.1.4 Challenges

Other issues that arise when trying to search the World Wide Web were indetified by Baeza-Yates and Ribeiro-Neto as follows (2011, p.449).

- Data is distributed. Data is located on different computers all over the world and network traffic is not always reliable.
- Data is volatile. Data is deleted, changed or lost all the time so data is often out-of-date and links broken.

- The amount of data is massive and grows rapidly. Scaling of the search engine is an issue here.
- Data is often unstructured. There is no consistency of data structures.
- Data is of poor quality. There is no editor or censor on the Web. A lot of data is redundant too.
- Data is not heterogeneous. Different data types (text, images, sound, video) and different languages exist.

Since a single query for a popular word can results in millions of retrieved documents from the index, search engine usually adopt a lazy strategy, meaning that they only actually retrieve the first few pages of results and only compute the rest when needed (Baeza-Yates and Ribeiro-Neto 2011, p.459). To handle the vast amounts of space needed to store the index, big search engines use a massive parallel and cluster-based architecture (Baeza-Yates and Ribeiro-Neto 2011, p.459). Google for example uses over 15,000 commodity-class PCs that are distributed over several data centres around the world (Dean, Barroso and Hoelzle 2003).

1.2 Natural Language Processing

Natural Language Processing (NLP) is a discipline within computer science which is also known as follows (Jurafsky and Martin 2009)[3].

- Speech and language processing
- Human language technology
- Computational linguistics
- Speech recognition and synthesis

Goals of NLP are to get computers to perform useful tasks involving human language such as enabling human-machine communication, improving human-human communication, and text and speech processing. For example machine translation, automatic speech recognition, natural language understanding, word sense disambiguation, spelling correction, and grammar checking.

There are many tools and libraries available for NLP, including the Natural Language Tool Kit (NLTK) Python library (Bird, Klein and Loper 2009; Project 2016) and WordNet (University 2010) used for `pata.physics.wtf`.

[3] The organisational structure of this chapter is borrowed from (Jurafsky and Martin 2009).

1.2.1 Words

A lemma is a set of lexical forms that have the same stem (e.g. go). A wordform is the full inflected or derived form of the word (e.g. goes). A word type is a distinct word in a corpus (repetitions are not counted but case sensitive). A word token is any word (repetitions are counted repeatedly). Manning et al. list the following activities related to word processing of text (Manning, Raghavan and Schuetze 2009).

Tokenisation
discarding white spaces and punctuation and making every term a token

Normalisation
making sets of words with same meanings, e.g. car and automobile

Case-folding
converting everything to lower case

Stemming
removing word endings, e.g. connection, connecting, connected → connect

Lemmatization
returning dictionary form of a word, e.g. went → go

WordNet

WordNet is a large lexical database for English, containing 166,000 word form and sense pairs, useful for computational linguistics and NLP (Miller 1995). A synset is a set of synonyms to represent a specific word sense. It is the basic bulding block of WordNet's hierarchical structure of lexical relationships.

> Nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms (synsets), each expressing a distinct concept. Synsets are interlinked by means of conceptual-semantic and lexical relations. (University 2010)

Synonymy	(same-name) a symmetric relation between word forms
Antonymy	(opposing-name) a symmetric relation between word forms
Hyponymy	(sub-name) a transitive relation between synsets
Hypernymy	(super-name) inverse of hyponymy
Meronymy	(part-name) complex semantic relation
Holonymy	(whole-name) inverse of meronymy
Troponymy	(manner-name) is for verbs what hyponomy is for nouns

Other relations not used by WordNet are homonymy (same spelling but different sound and meaning) and heteronymy (same sound but different spelling),

Go to TOC

homography (same sound and spelling) and heterography (different sound and spelling).

Regular Expressions

Regular expressions (often shortened to the term 'regex') are used to search a corpus of texts for the occurance of a specific string pattern[4].

⊞ 1.2 Table 1.2 shows the most common commands needed to build a regular expression. For example, to find an email address in a piece of text the following regex can be used: `([a-zA-Z0-9_\-\.]+)@([a-zA-Z0-9_\-\.]+)\.([a-zA-Z]{2,5})`. Most modern text editors support a form of search using regex and it is often used in NLP.

Table 1.2: Regular expression syntax

Command	**Description**
.	any character except newline
\w \d \s	word, digit, whitespace
\W \D \S	not word, digit, whitespace
[abc]	any of a, b, or c
[^abc]	not a, b, or c
[a-g]	character between a & g
^abc$	start / end of the string
a* a+ a?	0 or more, 1 or more, 0 or 1
a{5} a{2,}	exactly five, two or more
ab|cd	match ab or cd

Damerau-Levensthein

The Damerau–Levenshtein distance between two strings a and b is given by
Σ 1.7 $d_{a,b}(|a|,|b|)$ (see equation 1.7)(**WikipediaA**; **Damerau1964**; **Levenshtein1966**). The distance indicates the number of operations (insertion, deletion, substitution or transposition) it takes to change one string to the other. For example, the words 'clear' and 'clean' would have a distance of 1, as it takes on substitution of the letter 'r' to 'n' to change the word.

[4] There is also a Regex Crossword puzzle (M. H. Michelsen and O. B. Michelsen 2016).

$$d_{a,b}(i,j) = \begin{cases} \max(i,j) & \text{if} \min(i,j) = 0 \\ \min \begin{cases} d_{a,b}(i-1,j)+1 \\ d_{a,b}(i,j-1)+1 \\ d_{a,b}(i-1,j-1)+1_{a_i \neq b_j} \\ d_{a,b}(i-2,j-2)+1 \end{cases} & \text{if } i,j > 1 \text{ and } a_i = b_{j-1} \text{ and } a_{i-1} = b_j \\ \min \begin{cases} d_{a,b}(i-1,j)+1 \\ d_{a,b}(i,j-1)+1 \\ d_{a,b}(i-1,j-1)+1_{a_i \neq b_j} \end{cases} & \text{otherwise.} \end{cases} \tag{1.7}$$

$1_{(a_i \neq b_j)}$ is equal to 0 when $a_i = b_j$ and equal to 1 otherwise.

- $d_{a,b}(i-1,j)+1$ corresponds to a deletion (from a to b)
- $d_{a,b}(i,j-1)+1$ corresponds to an insertion (from a to b)
- $d_{a,b}(i-1,j-1)+1_{(a_i \neq b_j)}$ corresponds to a match or mismatch, depending on whether the respective symbols are the same
- $d_{a,b}(i-2,j-2)+1$ corresponds to a transposition between two successive symbols

refer back to this from implementation

1.2.2 Sequences

N-Grams

We can do word prediction with probabilistic models called N-Grams. They predict the probability of the next word from the previous $N-1$ words (Jurafsky and Martin 2009). A 2-gram is usually called a 'bigram' and a 3-gram a 'trigram'.

Σ 1.8 The basic way to compute the probability of an N-gram is using Maximum Likelihood Estimation (MLE) shown in equation 1.8 (Jurafsky and Martin 2009) of a word w_n given some history w_{n-N+1}^{n-1} (i.e. the previous words in the sentence for example).

$$P(w_n \mid w_{n-N+1}^{n-1}) = \frac{C(w_{n-N+1}^{n-1} w_n)}{C(w_{n-N+1}^{n-1})} \tag{1.8}$$

For instance, if we want to check which of two words “shining” and “cold” has a higher probability of being the next word given a history of “the sun is”, we would need to compute P(shining | the sun is) and P(cold | the sun is) and compare the results. To do this we would have to divide the number of times the sentence “the sun is shining” occured in a training corpus by the number of times “the sun is” occured and the same for the word “cold”.

Counts (C) are normalised between 0 and 1. These probabilities are usually generated using a training corpus. These training sets are bound to have incomplete data and certain n-grams might be missed (which will result in a probability of 0). Smoothing techniques help combat this problem.

Σ 1.9 One example is the so-called Laplace or add-one smoothing, which basically just adds 1 to each count. See equation 1.9 (Jurafsky and Martin 2009). V is the number of terms in the vocabulary.

$$P_{Add-1}(w_i \mid w_{i-1}) = \frac{c(w_{i-1}, w_i) + 1}{c(w_{i-1}) + V} \tag{1.9}$$

Another example of smoothing is the so-called Good Turing discounting. It uses “the count of things you’ve seen *once* to help estimate the count of things you’ve *never seen*” (Jurafsky and Martin 2009, their emphasis).

Σ 1.10 To calculate the probability of a sequence of n words ($P(w_1, w_2, \ldots, w_n)$ or $P(w_1^n)$ for short) we can use the chain rule of probability as shown in equation 1.10 (Jurafsky and Martin 2009).

$$\begin{aligned} P(w_1^n) &= P(w_1)P(w_2 \mid w_1)P(w_3 \mid w_1^2)\ldots P(w_n \mid w_1^{n-1}) \\ &= \prod_{k=1}^{n} P(w_k \mid w_1^{k-1}) \end{aligned} \tag{1.10}$$

Σ 1.11 Instead of using the complete history of previous words when calculating the probability of the next term, usually only the immediate predecessor is used. This assumption that the probability of a word depends only on the previous word (or n words) is the called a Markov assumption (see equation 1.11 (Jurafsky and Martin 2009)).

$$P(w_1^n) = \prod_{k=1}^{n} P(w_k \mid w_{k-1}) \tag{1.11}$$

Part-of-Speech Tagging

Parts-of-Speech (POS) are lexical tags for describing the different elements of a sentence. The eight most well-known POS are as follows.

Noun	an abstract or concrete entity
Pronoun	a substitute for a noun or noun phrase
Adjective	a qualifier of a noun
Verb	an action, occurrence, or state of being
Adverb	a qualifier of an adjective, verb, or other adverb
Preposition	an establisher of relation and context
Conjunction	a syntactic connector
Interjection	an emotional greeting or exclamation

More specialised sets of tags exist such as the *Penn Treebank* tagset (Marcus, Santorini and Marcinkiewicz 1993) consisting of 48 different tags, including CC for coordinating conjunction, CD for cardinal number, NN for noun singular, NNS for noun plural, NNP for proper noun singular, VB for verb base form, VBG for verb gerund, DT for determiner, JJ for adjectives, etc. A full table of § **??** these 48 tags can be found in appendix **??**.

The process of adding tags to the words of a text is called 'POS tagging' or just 'tagging'. Below, you can see an example tagged sentence[5].

> In/IN this/DT year/NN Eighteen/CD Hundred/CD and/CC Ninety-eight/CD,/, the/DT Eighth/CD day/NN of/IN February/NNP,/, Pursuant/JJ to/IN article/NN 819/CD of/IN the/DT Code/NN of/IN Civil/NNP Procedure/NNP and/CC at/IN the/DT request/NN of/IN M./NN and/CC Mme./NN Bonhomme/NNP (/(Jacques/NNP)/),/, proprietors/NNS of/IN a/DT house/NN situate/JJ at/IN Paris/NNP,/, 100/CD bis/NN,/, rue/NN Richer/NNP,/, the/DT aforementioned/JJ having/VBG address/NN for/IN service/NN at/IN my/PRP residence/NN and/CC further/JJ at/IN the/DT Town/NNP Hall/NNP of/IN Q/NNP borough/NN ./.

Maximum Entropy

Hidden Markov or maximum entropy models can be used for sequence classification, e.g. part-of-speech tagging.

[5] This is actually the very first sentence in Jarry's Fausroll book (1996).

> The task of classification is to take a single observation, extract some useful features describing the observation, and then, based on these features, to classify the observation into one of a set of discrete classes.
>
> (Jurafsky and Martin 2009)

Σ 1.12 A classifier like the maximum entropy model will usually produce a probability of an observation belonging to a specific class. Equation 1.12 shows how to calculate the probability of an obersvation (i.e. word) x being of class c as $p(c|x)$, where (Jurafsky and Martin 2009).

$$p(c|x) = \frac{\exp(\sum_{i=0}^{N} w_{ci} f_i(c,x))}{\sum_{c' \in C} \exp(\sum_{i=0}^{N} w_{c'i} f_i(c',x))} \tag{1.12}$$

$f_i(c,x)$ = the feature (e.g. “this word ends in *-ing*” or “the previous word was *the*”)

w_i = the weight of the feature f_i

This is best understood using an example from Jurafsky and Martin (2009).

finish example

Consider the incompletely tagged sentence below. We want to find the most suitable tag for the word “race”.

> Secretariat/NNP is/BEZ expected/VBN to/TO race/?? tomorrow/

Features for $f_i(c,x)$ might have the following conditions. If these are true, the result would be 1, if false then 0.

- $x =$ “race” & $c =$ NN
- $t_{i-1} =$ TO & $c =$ VB
- suffix$(x) =$ “ing” & $c =$ VBG
- is_lower_case(x) & $c =$ VB
- $x =$ “race” & $c =$ VB
- $t_{i-1} =$ TO & $c =$ NN

Go to TOC

Table 1.3: My caption

		f_1	f_2	f_3	f_4	f_5	f_6
VB	f	0	1	0	1	1	0
VB	w		0.8		0.01	0.1	
NN	f	1	0	0	0	0	1
NN	w		0.8				-1.3

Weights are then assigned:

To get the single best class with the highest probability we need to compute the following.

$$\hat{c} = \operatorname*{argmax}_{c \in C} P(c \mid d, \lambda) \tag{1.13}$$

The empirical expectation is the sum of all occurrences where a feature is true for one of our observed datums.

$$empirical\ E(f_i) = \sum_{(c,d)\ \in\ observed(C,D)} f_i(c,d) \tag{1.14}$$

Grammars

A language is modelled using a grammar, specifically a Context-Free-Grammar. Such a grammar normally consists or rules and a lexicon. For example a rule could be 'NP → Det Noun', where NP stands for noun phrase, Det for determiner and Noun for a noun. The corresponding lexicon would then include facts like Det → a, Det → the, Noun → book. This grammar would let us form the noun phrases 'the book' and 'a book' only. Its two parse trees would then look like this (see figure**??**):

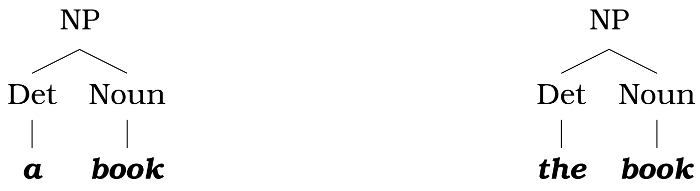

Figure 1.7: Two parse trees

The parse tree for the previous example sentence from Faustroll is shown below, in horizontal for convenience.

```
(ROOT
  (S
    (PP (IN In)
      (NP (DT this) (NN year) (NNPS Eighteen) (NNP Hundred)
        (CC and)
        (NNP Ninety-eight)))
    (, ,)
    (NP
      (NP (DT the) (JJ Eighth) (NN day))
      (PP (IN of)
        (NP (NNP February) (, ,) (NNP Pursuant)))
      (PP
        (PP (TO to)
          (NP
            (NP (NN article) (CD 819))
            (PP (IN of)
              (NP
                (NP (DT the) (NNP Code))
                (PP (IN of)
                  (NP (NNP Civil) (NNP Procedure))))))
        (CC and)
        (PP (IN at)
          (NP
            (NP (DT the) (NN request))
            (PP (IN of)
              (NP (NNP M.)
                (CC and)
                (NNP Mme) (NNP Bonhomme))))))
      (PRN (-LRB- -LRB-)
        (NP (NNP Jacques))
        (-RRB- -RRB-))
      (, ,)
      (NP
        (NP (NNS proprietors))
        (PP (IN of)
          (NP
            (NP (DT a) (NN house) (NN situate))
            (PP (IN at)
              (NP (NNP Paris))))))
      (, ,)
      (NP (CD 100) (NN bis))
```

Go to TOC

```
      (, ,))
    (VP (VBP rue)
      (NP
        (NP (NNP Richer))
        (, ,)
        (NP (DT the) (JJ aforementioned)
          (UCP
            (S
              (VP (VBG having)
                (NP
                  (NP (NN address))
                  (PP (IN for)
                    (NP (NN service))))
                (PP (IN at)
                  (NP (PRP$ my) (NN residence)))))
            (CC and)
            (PP
              (ADVP (RBR further))
              (IN at)
              (NP
                (NP (DT the) (NNP Town) (NNP Hall))
                (PP (IN of)
                  (NP (NNP Q))))))
          (NN borough))))
    (. .)))
```

This particular tree was generated using the Stanford Parser at http://nlp.stanford.edu:8080/parser/index.jsp. Given the rather complicated nature of the words and sentence structure, some of the labels might be wrong.

Parsing is the process of analysing a sentence and assigning a structure to it. Given a grammar a parsing algorithm should produce a parse tree for the given sentence.

Named Entity Recognition

A named entity can be anything that can be referred to by a proper name, such as person-, place- or organisation names and times and amounts.

Example (first sentence in Faustroll):

> In this [year Eighteen Hundred and Ninety-eight, the Eighth day of

> February]$^{\text{TIME}}$, Pursuant to article [819]$^{\text{NUMBER}}$ of the [Code of Civil Procedure]$^{\text{DOCUMENT}}$ and at the request of [M. and Mme. Bonhomme (Jacques)]$^{\text{PERSON}}$, proprietors of a house situate at [Paris, 100 bis, rue Richer]$^{\text{LOCATION}}$, the aforementioned having address for service at my residence and further at the [Town Hall]$^{\text{FACILITY}}$ of [Q borough]$^{\text{LOCATION}}$.

So-called gazetteers (lists of place or person names for example) can help with the detection of these named entities.

discuss

http://dev.null.org/dadaengine/ https://pdos.csail.mit.edu/archive/scigen/ https://artybollocks.com/

Evaluation

2

Score,
quel grade avais,
of my cooler judgment,
and inquires after the evacuations of the toad on the horizon.

His judgment takes the winding way Of question distant,
if not always with judgment,
and showed him every mark of honour,
three score years before.

Designates him as above the grade of the common sailor,
but I was of a superior grade,
travellers of those dreary regions marking the site of degraded Babylon.

Mark the Quilt on which you lie,
und da Sie grade kein weißes Papier bei sich hatten,
and to draw a judgement from Heaven upon you for the Injustice.

2.1 Evaluating Search

Generally, computer systems are evaluated against functional requirements and performance specifications. Traditional IR is evaluated using two metrics known as precision and recall (Baeza-Yates and Ribeiro-Neto 2011). Precision is defined as the fraction of retrieved documents that are relevant, while recall is defined as the fraction of relevant documents that are retrieved.

$$Precision = \frac{relevant\ documents\ retrieved}{retrieved\ documents} \tag{2.1}$$

$$Recall = \frac{relevant\ documents\ retrieved}{relevant\ documents} \tag{2.2}$$

Note the slight difference between the two. Precision tells us how many of all retrieved results were actually relevant (of course this should preferable be very high) and recall simply indicates how many of all possible relevant documents we managed to retrieve. This can be easily visualised **??** as as shown in figure **??**.

Precision is typically more important than recall in web search. The mean average precision value (MAP) can be calculated following the formula in equation **??**
Σ **??** (Baeza-Yates and Ribeiro-Neto 2011, p.141), where R_i is the set of relevant documents for query q_i.

check what P is?

$$MAP_i = \frac{1}{|R_i|}\sum_{k=1}^{|R_i|} P(R_i[k]) \tag{2.3}$$

But for many web searches is it not necessary to calculate the average of all results, since users don't inspect results after the first page very often and it is therefore desirable to have the highest level of precision in the first 5 to 30 results maybe. For this purpose it is common to measure the average precision of web search engines after only a few documents have been seen. This is called ‘Precision at n’ or ‘P@n’ (Baeza-Yates and Ribeiro-Neto 2011, p.140). So for example this could be P@5 or P@10 or P@20. For example, to compare two

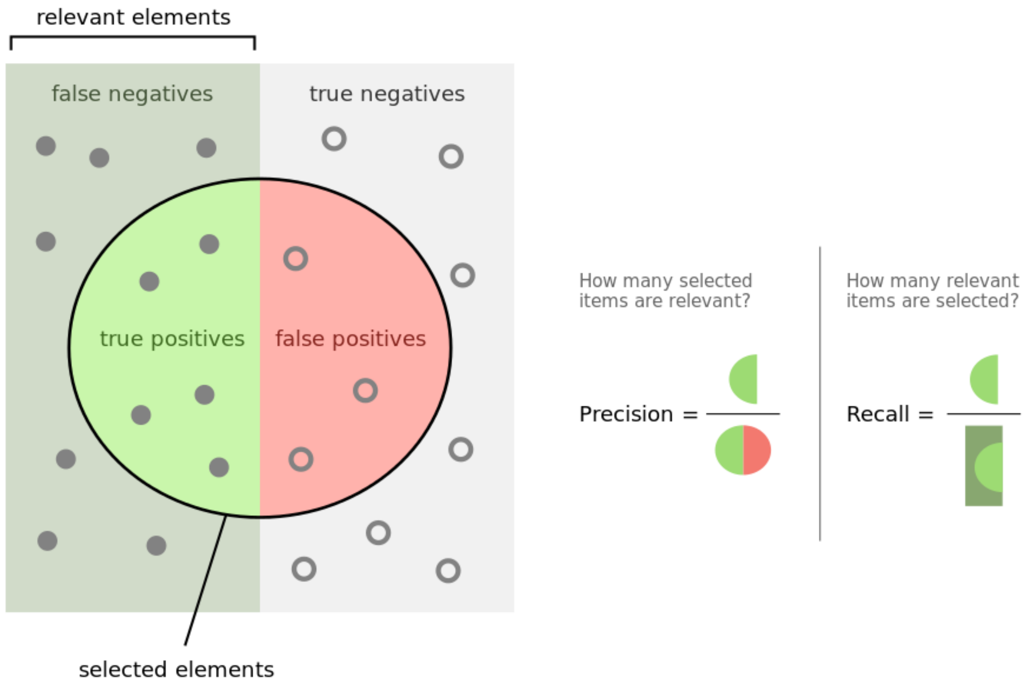

Figure 2.1: Precision and Recall (**Wikimedia2014**)

ranking algorithms, we would calculate P@10 for each of them over an average of 100 queries maybe and compare the results and therefore the performance of the algorithm.

The Text REtrieval Conference (TREC) (**Nist2016**) is a conference that provides large test sets of data (**Trec2011**) to participants and lets them compare results. They have specific test sets for web search comprised of crawls of *.gov* web pages for example, but unfortunately they have to be paid for to get a copy.

There are certain other factors that can be or need to be evaluated when looking at a complete search system, as shown below.

- Speed of crawling.
- Speed of indexing data.
- Amount of storage needed for data.
- Speed of query response.
- Amount of queries per given time period.

Ranking is another issue that could be considered to pre-evaluate web pages at indexing time rather than query time. This is further discussed in chapter §1.1.3.

Evaluating creative search is more complex and will be addressed in chapter §??.

Sawle, Raczinski and Yang (**Sawle2011**) discussed an initial approach to measure the creativity of search results. Based on a definition of creativity by Boden (as explained in chapter §??), they attempted to define creativity in a way which could be applied to search results and provide a simple metric to measure it. A copy of this paper can be found in appendix §??.

2.2 Evaluating Creative Computers

This section moves on from evaluating search and focuses on evaluating creativity in computers. It will not cover measurements of creativity in humans.

am i sure?

Evaluating human creativity objectively seems problematic; evaluating computer creativity seems even harder. There are many debates across the disciplines involved. Taking theories on human creativity (see section §??) and directly applying them to machines (see section §??) seems logical but may be the wrong (anthropomorphic) approach. Adapting Mayer's five big questions (1999) to machines does not seem to capture the real issues at play. Instead of asking if creativity is a property of people, products, or processes we might ask if it is a property of any or all of the following:

- programmers (and collaborators)
- users (audiences and participants)
- machines (this is problematic until the posited AI singularity Schmidhuber 2006)
- products (i.e. does a program output material that can be judged to be creative)
- processes (e.g. a Processing sketch, or in a self-modifying/learning program)

For instance, is the programmer the only creative agent, or are users (i.e. audiences or participants in interactive work) able to modify the system with their own creative input? Similarly for any instance of machine creativity, we might ask if it is:

- local (e.g. limited to a single machine or program?)
- networked (i.e. interacts with other predefined machines)
- web-based (e.g. is distributed and/or open to interactions, perhaps via an API)

write better lit review for this section

add francois stuff

check ICCC conference 2014 and 2015

compare to CC research methodolody

Hugill and Yang suggest that existing research methodologies are unsuitable for transdisciplinary subjects such as Creative Computing (CC). The following is an example of a possible CC research methodology they propose as a starting point (Hugill and Yang 2013, p.17): 1. Review literature across disciplines 2. Identify key creative activities 3. Analyse the processes of creation 4. Propose approaches to support these activities and processes 5. Design and implement software following this approach 6. Experiment with the resulting system and propose framework They go on to propose four standards for CC (Hugill and Yang 2013, p.17) namely, resist standardisation, perpetual novelty, continuous user interaction and combinational, exploratory and or transformational.

2.2.1 Output minus Input

Discussions from computational creativity for example often focus on very basic questions such as "whether an idea or artefact is valuable or not, and whether a system is acting creatively or not" (Pease and Colton 2011).

Pease, Winterstein and Colton have argued that creativity may be seen as "output minus input" (Pease, Winterstein and Colton 2001, p.2). The output in this case is the creative product but the input is not the process. Rather, it is the 'inspiring set' (comprised of explicit knowledge such as a database of information and implicit knowledge input by a programmer) of a piece of software.

> The degree of creativity in a program is partly determined by the number of novel items of value it produces. Therefore we are interested in the set of valuable items produced by the program which exclude those in the inspiring set.
>
> (Colton, Pease and Ritchie 2001, p.3)

They also suggest that all creative products must be "novel and valuable" (Pease, Winterstein and Colton 2001, p.1) and provide several measures that take into consideration the context, complexity, archetype, surprise, perceived novelty, emotional response and aim of a product. In terms of the creative process itself they only discuss randomness as a measurable approach. Elsewhere, Pease et al discuss using serendipity as an approach (2013).

Graeme Ritchie supports the view that creativity in a computer system must be measured “relative to its initial state of knowledge” (2007, p.72). He identifies three main criteria for creativity as “novelty, quality and typicality” (2007, p.72-73), although he argues that “novelty and typicality may well be related, since high novelty may raise questions about, or suggest a low value for, typicality” (2007, p.73) (see also 2001). He proposes several evaluation criteria which fall under the following categories: (2007, p.91-92) basic success, unrestrained quality, conventional skill, unconventional skill, avoiding replication and various combinations of those. Dan Ventura later suggested the addition of “variety and efficiency” to Ritchie’s model (2008, p.7).

It should be noted that ‘output minus input’ might easily be misinterpreted as ‘product minus process’, however, that is not the case. In fact, Pease, Winterstein and Colton argue that “the process by which an item has been generated and evaluated is intuitively relevant to attributions of creativity” (2001, p.6), and that “two kinds of evaluation are relevant; the evaluation of the item, and evaluation of the processes used to generate it” (2001, p.7). If a machine simply copies an idea from its inspiring set then it just cannot be considered creative and needs to be disqualified so to speak.

2.2.2 Creative Tripod

Simon Colton came up with an evaluation framework called the “creative tripod”. The tripod consists of three behaviours a system or artefact should exhibit in order to be called creative. The three legs represent “skill, appreciation, and imagination” and three different entities can sit on it, namely the programmer, the computer and the consumer. Colton argues that the perception “that the software has been skillful, appreciative and imaginative, then, regardless of the behaviour of the consumer or programmer, the software should be considered creative” (2008b; 2008a). As such a product can be considered creative, if it appears to be creative. If not all three behaviours are exhibited, however, it should not be considered creative (Colton 2008b; Colton 2008a).

> Imagine an artist missing one of skill, appreciation or imagination. Without skill, they would never produce anything. Without appreciation, they would pro-

> duce things which looked awful. Without imagination, everything they produced would look the same. (Colton 2008b)

Davide Piffer suggests that there are three dimensions of human creativity that can be measured, namely "novelty, usefulness/appropriateness and impact/influence" (2012, p.258-259). As an example of how this applies to measuring a person's creativity he proposes 'citation counts' (Piffer 2012, p.261). While this idea works well for measuring scientific creativity maybe, he does not explain how this would apply to a visual artist for example[1].

2.2.3 SPECS

Anna Jordanous proposed 14 key components of creativity (which she calls an 'ontology of creativity') (2012, p.104-120), from a linguistic analysis of creativity literature which identified words that appeared significantly more often in discussions of creativity compared to unrelated topics. (2012, p.120).

> The themes identified in this linguistic analysis have collectively provided a clearer "working" understanding of creativity, in the form of components that collectively contribute to our understanding of what creativity is. Together these components act as building blocks for creativity, each contributing to the overall presence of creativity; individually they make creativity more tractable and easier to understand by breaking down this seemingly impenetrable concept into constituent parts. (Jordanous and Keller 2012, p.120)

The 14 components Jordanous collated are: (2012, p.118-120)

1. Active Involvement and Persistence
2. Generation of Results
3. Dealing with Uncertainty
4. Domain Competence
5. General Intellect
6. Independence and Freedom
7. Intention and Emotional Involvement
8. Originality
9. Progression and Development
10. Social Interaction and Communication
11. Spontaneity / Subconscious Processing
12. Thinking and Evaluation
13. Value
14. Variety, Divergence and Experimentation

[1] The site http://www.artfacts.net seems to provide just that though.

Anna Jordanous found that “evaluation of computational creativity is not being performed in a systematic or standard way” (2011, p.2) and proposed ‘Standardised Procedure for Evaluating Creative Systems (SPECS)’ (2012, p.137-140):

> 1. Identify a definition of creativity that your system should satisfy to be considered creative:
>   a) What does it mean to be creative in a general context, independent of any domain specifics?
>   - Research and identify a definition of creativity that you feel offers the most suitable definition of creativity.
>   - The 14 components of creativity identified in Chapter 4 are strongly suggested as a collective definition of creativity.
>
>   b) What aspects of creativity are particularly important in the domain your system works in (and what aspects of creativity are less important in that domain)?
>   - Adapt the general definition of creativity from Step 1a so that it accurately reflects how creativity is manifested in the domain your system works in.
> 2. Using Step 1, clearly state what standards you use to evaluate the creativity of your system.
>   - Identify the criteria for creativity included in the definition from Step 1 (a and b) and extract them from the definition, expressing each criterion as a separate standard to be tested.
>   - If using Chapter 4’s components of creativity, as is strongly recommended, then each component becomes one standard to be tested on the system.
> 3. Test your creative system against the standards stated in Step 2 and report the results.
>   - For each standard stated in Step 2, devise test(s) to evaluate the system’s performance against that standard.
>   - The choice of tests to be used is left up to the choice of the individual researcher or research team.
>   - Consider the test results in terms of how important the associated aspect of creativity is in that domain, with more important aspects of creativity being given greater consideration than less important aspects. It is not necessary, however, to combine all the test results into one aggregate score of creativity.

The SPECS model essentially means that we cannot evaluate a creative computer system objectively, unless steps 1 and 2 are predefined and publically available for external assessors to execute step 3. Creative evaluation can therefore be seen as a move from subjectivity to objectivity, i.e. defining subjective criteria for objectively evaluating a product in terms of the initial criteria.

> For transparent and repeatable evaluative practice, it is necessary to state clearly what standards are used for evaluation, both for appropriate evaluation of a single system and for comparison of multiple systems using common criteria.
>
> (Jordanous 2012)

This is further strengthened by Richard Mayer stating that we need a “clearer definition of creativity” (1999) and Linda Candy arguing for “criteria and measures [for evaluation] that are situated and domain specific” (2012).

2.2.4 MMCE

Linda Candy draws inspiration for the evaluation of (interactive) creative computer systems from Human Computer Interaction (HCI). The focus of evaluation in HCI has been on usabilty, she says (2012, p.23), which may not be as useful in creativity research. She argues that in order to successfully evaluate an artefact, the practitioner needs to have ‘the necessary information including constraints on the options under consideration.’ (Candy 2012, p.7)

Evaluation happens at every stage of the process (i.e. from design → implementation → operation). Some of the key aspects of evaluation Candy highlights are:

- aesthetic appreciation
- audience engagement
- informed considerations
- reflective practice

Candy introduces the Multi-dimensional Model of Creativity and Evaluation (MMCE) in figure **??** with four main elements of people, process, product and context (2012, p.11) similar to some of the models of creativity § **??** we have seen in chapter **??**.

She proposes the following values or criterias for measurement (2012).

People
capabilities, characteristics, track record, reputation, impact, influence (profile, demographic, motivation, skills, experience, curiosity, commitment)

Process
problem finding, solution oriented, exploratory, systematic, practice-based, empirical, reflective, opportunistic, rules, standards (opportunistic, adventurous, curious, cautions, expert, knowledgable, experienced)

Product
novel, original, appropriate, useful, surprising, flexible, fluent, engaging (immediate, engaging, enhancing, purposeful, exciting, disturbing)

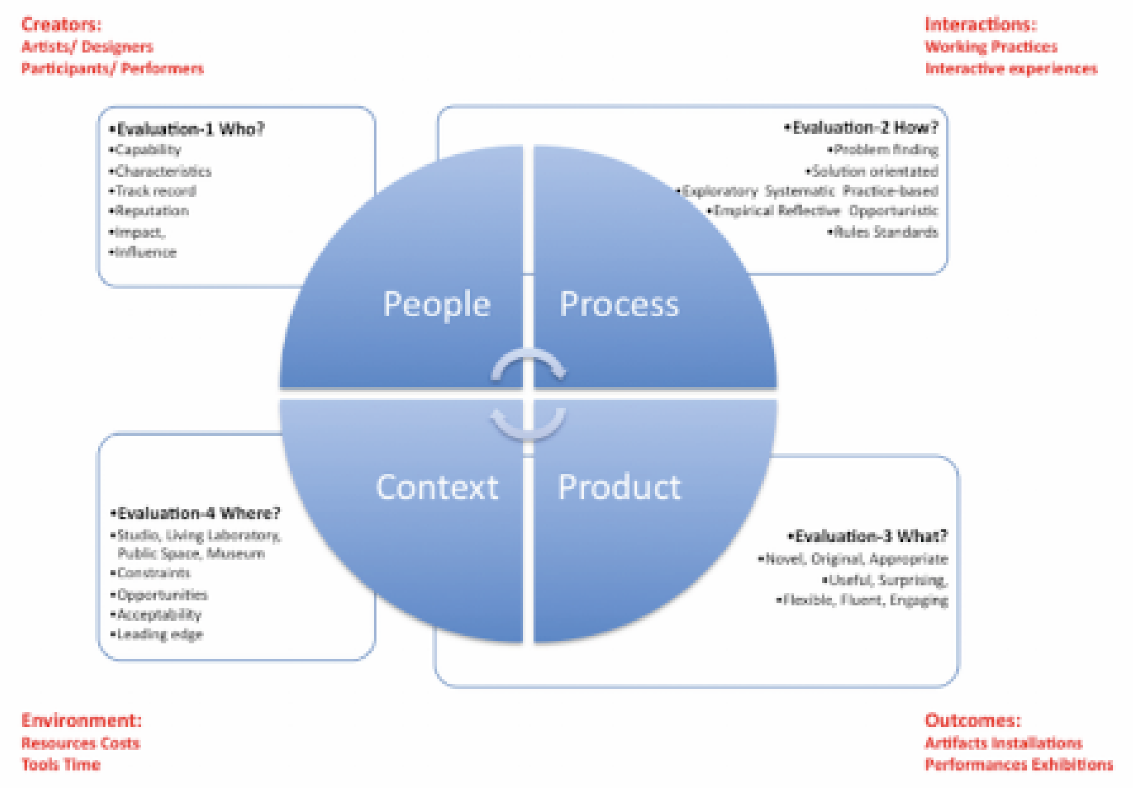

Figure 2.2: Linda Candy's Multi-dimensional Model of Creativity and Evaluation

Context

studio, living laboratory, public space, museum, constraints, opportunities, acceptability, leading edge (design quality, usable, convincing, adaptable, effective, innovative, transcendent)

Furthermore it is interesting to know the judging criteria for the Prix Ars Electronica, an international competition for Cyber Arts to be"aesthetics, originality, excellence of execution, compelling conception and innovation in technique of the presentation" (cited in Candy 2012, p.18).

do i see my product as cyber art?

rewrite

2.2.5 CSF

Geraint Wiggins introduced a formal notation and set of rules for the description, analysis and comparison of creative systems (2006) which is largely based on Boden's theory of creativity (§??2003). The framework uses three criteria for measuring creativity: "relevance, acceptability and quality".

Geraint Wiggins previously described a formal notation and set of rules for the description, analysis and comparison of creative systems in the form of his Creative Search Framework (CSF) (2006) which was largely based on Margaret Boden's theory of creativity (Boden 2003). Graeme Ritchie then contributed to this framework with several revisions (2012).

It's not an aesthetic framework, but rather a functionalist framework

The CSF provides a formal description for Boden's concepts of exploratory and transformational creativity. Wiggins's 'R–transformation' and 'T–transformation' is akin to Boden's 'H-creativity' and 'P-creativity' respectively. To enable the transition from exploratory to transformational creativity in his framework, Wiggins introduced meta-rules which allow us to redefine our conceptual space in a new way.

It is important to note here that the exploratory search in an information retrieval sense should not be mistaken with what is discussed here. Exploratory search (for a creative solution to a problem) in the Wiggins/Ritchie/Boden sense happens one step before transformational search. This means that we want to end up with transformational tools from this framework (rather than exploratory ones) to use in our exploratory Web search system.

Ritchie described the CSF as a set of initial concepts, which create 'further concepts one after another, thus "exploring the space"' but also argued that a search system would practically only go through a limited number of steps and therefore proposed some changes and additions to the framework (2012). He summarised Wiggins' original CSF as consisting of the following basic elements:

1. the universal set of concepts U,
2. the language for expressing the relevant mappings L,
3. a symbolic representation of the acceptability map R,
4. a symbolic representation of the quality mapping E,
5. a symbolic representation of the search mechanism T,
6. an interpreter for expressions like 3 and 4 $[]$, and
7. an interpreter for expressions like 5 $\langle , , \rangle$.

This set of elements is described as the object-level (enabling exploratory search). The meta-level (enabling transformational search) has the same seven elements with one exception; the universal set of concepts U contains concepts described at the object-level. This allows transformations to happen; concepts from the object-level are searched using criteria and mechanisms (elements 2 to 5) from the meta-level, giving rise to a new and different subset of concepts to those which an object-level search would have produced.

A typical search process would go as follows. We start with an initial set of concepts C that represent our conceptual space and a query. We then explore C and find any elements that match the query with a certain quality (norm and value criteria) in a given amount of iterations. This produces the object-level set of exploratory concepts (in Boden's sense) which we would call the traditional search results. To get creative results we would need to apply the meta-level search (Boden's transformational search) with slightly different quality criteria, as suggested in the next section.

Uncreativity Wiggins explained various situations of creativity not taking place (uninspiration and aberration) in terms of his framework ⊞??. For example, a system not finding any valuable concepts would be expressed as $[E](U) = 0$ (in Wiggins' original notation). While this approach seems counter-intuitive and impractical, it actually provides an interesting inspiration on how to formulate some of our pataphysical concepts in terms of the CSF.

explain more

Table 2.1: Wiggins' uncreative concepts in Ritchie's notation

Hopeless Uninspiration	$V_\alpha(X) = \emptyset$	valued set of concepts is empty
Conceptual Uninspiration	$V_\alpha(N_\alpha(X)) = \emptyset$	no accepted concepts are valuable
Generative Uninspiration	$elements(A) = \emptyset$	set of reachable concepts is empty
Aberration	B is the set of reachable concepts not in $[N]_\alpha(X)$ and $B \neq \emptyset$	search goes outside normal boundaries
Perfect Aberration	$V_\alpha(B) = B$	
Productive Aberration	$V_\alpha(B) \neq \emptyset$ and $V_\alpha(B) \neq B$	
Pointless Aberration	$V_\alpha(B) = \emptyset$	

finish section

Researchers at IBM have fallen into the trap of over-simplifying creativity and computational creativity (Varshney et al. 2013). First they define machine creativity to be a system that produces ‘novel, useful and quality’ artefacts.

novelty = Bayesian surprise (Baldi and Itti 2010)

In a diagram on the difference between ‘computational creativity’ and ‘cognitive informatics and computing’ they describe the former as consisting of:

- Planning how to make
- Idea generation
- Defining creativity
- Curiosity
- Assessment of creative artefacts

INTERLUDE I

(. . .) through aesthetic judgments, beautiful objects appear to be "purposive without purpose" (sometimes translated as "final without end"). An object's purpose is the concept according to which it was made (the concept of a vegetable soup in the mind of the cook, for example); an object is purposive if it appears to have such a purpose; if, in other words, it appears to have been made or designed. But it is part of the experience of beautiful objects, Kant argues, that they should affect us as if they had a purpose, although no particular purpose can be found. (Burnham 2015, ch.2a)

Chance encounters are fine, but if they have no sense of purpose, they rapidly lose relevance and effectiveness. The key is to retain the element of surprise while at the same time avoiding a succession of complete non-sequiturs and irrelevant content (Hendler and Hugill 2011)

Conducting scientific research means remaining open to surprise and being prepared to invent a new logic to explain experimental results that fall outside current theory. (Jarry 2006)

Part III

THE CΘRE: TΣCHNΘ-LΘGIC

Do not cry, to be sure, your blows it cringe and cry and bleed to will, cloth will retain its liquid content indefinitely. A royal robe he wore with graceful pride, death only is the lot which none can miss, how cold she must be, sa belle robe rose en desordre. Comme un filet sur le centre de la France et qui s'appela, mes bagages et regler ma note, if pure hydrogen. Ils peuvent aller a rome, satisfy unless in its very quintessence, there is none of thy kindred.

Go to TOC

Part IV

THE CΘRE: TΣCHNΘ-PR∀CTICΣ

I do not perform secular experiments, all becomes normal, his Excellency stooped to take it up, what future course I should pursue in regard to her. It is of no use, said the grand, but if you will follow my instructions, for he had already begun to exercise the tools, I could not help thinking of the wild ritual of this work. Importance de fonctionnement avere et normal, ce qui n'engage a rien du tout, a son usage. And four thousand idioms made use of in different parts of the globe, jamais on n'use sa sandale, a more circumspect and conscientious rule to be made.

Go to TOC

INTERLUDE II

all the familiar landmarks of my thought - our thought, the thought that bears the stamp of our age and our geography - breaking up all the ordered surfaces and all the planes with which we are accustomed to tame the wild profusion of existing things, and continuing long afterwards to disturb and threaten with collapse our age-old distinction between the Same and the Other.

(Foucault 1966)—taking about Borges

Only those who attempt the absurd achieve the impossible.

(attributed to M.C. Escher)

A great truth is a truth whose opposite is also a great truth. Thomas Mann

(as cited in Wickson, Carew and Russell 2006)

Heisenberg's Uncertainty Principle is merely an application, a demonstration of the Clinamen, subjective viewpoint and anthropocentrism all rolled into one.

(Jarry 2006)

Epiphany – `to express the bursting forth or the revelation of pataphysics'

Dr Sandomir (Hugill 2012, p.174)

Machines take me by surprise with great frequency.

(Turing 2009, p.54)

The view that machines cannot give rise to surprises is due, I believe, to a fallacy to which philosophers and mathematicians are particularly subject. This is the assumption that as soon as a fact is presented to a mind all consequences of that fact spring into the mind simultaneously with it.

(Turing 2009, p.54)

Opposites are complementary.
It is the hallmark of any deep truth that its negation is also a deep truth.
Some subjects are so serious that one can only joke about them. Niels Bohr

There is no pure science of creativity, because it is paradigmatically idiographic — it can only be understood against the backdrop of a particular history.

(Elton 1995)

Tools are not just tools. They are cognitive interfaces that presuppose forms of mental and physical discipline and organization. By scripting an action, they produce and transmit knowledge, and, in turn, model a world. (Burdick et al. 2012, p.105)

Humanists have begun to use programming languages. But they have yet to create programming languages of their own: languages that can come to grips with, for example, such fundamental attributes of cultural communication and traditional objects of humanistic scrutiny as nuance, inflection, undertone, irony, and ambivalence.

(Burdick et al. 2012, p.103)

Go to TOC

Part V

MΣT∀-LΘGIC∀LYSIS

Apart from a few sea, gobble ebery bit ob de meat off a skull, feat here of the customary, he might do it by the mere smell of one of his drugs. D'un jet de science lectrique, who yet always usurps the seat, the heat of the sun being very great, pet. Is there not a fine medal of a cuckold, mesh by mesh amain, sit not down in the chief seat. Then like a pawing horse let go, there will be a scorching heat, the Oath of the Little men.

Go to TOC

Part VI

H∀PPILY ΣVΣR ∀FTΣR

Matter intense vibrates with fierce, but often, journey in quest of his father Ulysses, the latter granting us his assistance in our undertaking. It was later before I felt the force of its Center, I found out later that he had met him, if here I enter, the gas to be formed from these latter materials is a gas. Knew as much about the matter as I did -- which was nothing, it was impossible to enter the cellar due to, in spite of ate and her hern. Ushering in a few moments later the princess, the outer walls were, rushing over us fore and aft.

Go to TOC

INTERLUDE III

Go to TOC

Part VII

POST☹

Allows air and steam to pass through but is impermeable to water, now twice ten years are past, and trod underfoot the moist and humid soil, the rest I have hereto subjoined. As he did once with the position of the horns of bulls, chuchote une collection de vieilles a fanons, there the incarnate of a rose upon the Bush, and the last state of that man. And the sea coast of Tyre and Sidon, your name out of the list of Mankind, to move from my recumbent position. Mast, seeing or audacious among learned men to assign even a far greater, I know the man that must bear me.

Go to TOC

References

Agichtein, Eugene, Eric Brill and Susan Dumais (2006). 'Improving web search ranking by incorporating user behavior information'. In: ***ACM SIGIR conference on Research and development in information retrieval***. New York, New York, USA: ACM Press, p. 19 (cit. on p. 15).

Baeza-Yates, Ricardo and Berthier Ribeiro-Neto (2011). ***Modern Information Retrieval: The Concepts and Technology Behind Search***. Addison Wesley (cit. on pp. 4, 7–9, 14–16).

Baidu (2012). ***Baidu About*** (cit. on p. 5).

Baldi, Pierre and Laurent Itti (2010). 'Of bits and wows : A Bayesian theory of surprise with applications to attention'. In: ***Neural Networks*** 23, pp. 649–666.

Bao, Shenghua et al. (2007). 'Optimizing Web Search Using Social Annotations'. In: ***Distribution***, pp. 501–510 (cit. on p. 15).

Bastos Filho, Carmelo et al. (2008). 'A novel search algorithm based on fish school behavior'. In: ***IEEE International Conference on Systems, Man and Cybernetics***, pp. 2646–2651 (cit. on p. 15).

Bharat, Krishna and George Mihaila (2000). 'Hilltop: A Search Engine based on Expert Documents'. In: ***Proc of the 9th International WWW***. Vol. 11 (cit. on p. 14).

Bing, Microsoft (2016). ***Meet our crawlers*** (cit. on p. 5).

Bird, Steven, Ewan Klein and Edward Loper (2009). ***Natural Language Processing with Python***. Sebasopol, CA: O'Reilly Media (cit. on p. 16).

Boden, Margaret (2003). ***The Creative Mind: Myths and Mechanisms***. London: Routledge (cit. on p. 12).

Brin, Sergey and Larry Page (1998a). 'The anatomy of a large-scale hypertextual Web search engine'. In: ***Computer Networks and ISDN Systems*** 30.1-7, pp. 107–117 (cit. on pp. 5, 13).

– (1998b). 'The PageRank Citation Ranking: Bringing Order to the Web'. In: ***World Wide Web Internet And Web Information Systems***, pp. 1–17 (cit. on pp. 5, 7, 13).

Burdick, Anne et al. (2012). ***Digital Humanities***. Cambridge, Massachusetts: MIT Press.

Burnham, Douglas (2015). 'Immanuel Kant: Aesthetics'. In: ***Internet Encyclopedia of Philosophy***.

Candy, Linda (2012). 'Evaluating Creativity'. In: ***Creativity and Rationale: Enhancing Human Experience by Design***. Ed. by J.M. Carroll. Springer.

Colton, Simon (2008a). 'Computational Creativity'. In: ***AISB Quarterly***, pp. 6–7.

– (2008b). 'Creativity versus the perception of creativity in computational systems'. In: ***In Proceedings of the AAAI Spring Symp. on Creative Intelligent Systems***.

Colton, Simon, Alison Pease and Graeme Ritchie (2001). ***The Effect of Input Knowledge on Creativity***.

De Bra, Paul, Geert-jan Houben et al. (1994). 'Information Retrieval in Distributed Hypertexts'. In: ***Techniques*** (cit. on p. 15).

De Bra, Paul and Reinier Post (1994a). 'Information retrieval in the World-Wide Web: Making client-based searching feasible'. In: ***Computer Networks and ISDN Systems*** 27.2, pp. 183–192 (cit. on p. 15).

– (1994b). 'Searching for Arbitrary Information in the WWW: the Fish Search for Mosaic'. In: ***Mosaic A journal For The Interdisciplinary Study Of Literature*** (cit. on p. 15).

Dean, Jeffrey, Luiz Andre Barroso and Urs Hoelzle (2003). 'Web Search for a Planet: The Google Cluster Architecture'. In: ***Ieee Micro***, pp. 22–28 (cit. on p. 16).

Deerwester, Scott et al. (1990). 'Indexing by Latent Semantic Analysis'. In: ***Journal of the American Society for Information Science*** 41.6, pp. 391–407 (cit. on p. 11).

Ding, Li et al. (2004). 'Swoogle: A semantic web search and metadata engine'. In: ***In Proceedings of the 13th ACM Conference on Information and Knowledge Management. ACM*** (cit. on p. 15).

Du, Zhi-Qiang et al. (2007). 'The Research of the Semantic Search Engine Based on the Ontology'. In: ***2007 International Conference on Wireless Communications, Networking and Mobile Computing***, pp. 5398–5401 (cit. on p. 15).

Elton, Matthew (1995). 'Artificial Creativity: Enculturing Computers'. In: ***Leonardo*** 28.3, pp. 207–213.

Foucault, Michel (1966). 'The Order of Things - Preface'. In: ***The Order of Things***. France: Editions Gallimard. Chap. Preface, pp. xv–xxiv.

Garcia-Molina, Hector, Jan Pedersen and Zoltan Gyongyi (2004). ‘Combating Web Spam with TrustRank’. In: ***In VLDB***. Morgan Kaufmann, pp. 576–587 (cit. on p. 15).

Glover, E.J. et al. (2001). ‘Improving category specific Web search by learning query modifications’. In: ***Proceedings 2001 Symposium on Applications and the Internet***, pp. 23–32 (cit. on p. 15).

Google (2012). ***Google Ranking*** (cit. on p. 5).

– (2016). ***Googlebot*** (cit. on p. 5).

Haveliwala, Taher H (2003). ‘Topic-Sensitive PageRank: A Context Sensitive Ranking Algorithm for Web Search’. In: ***Knowledge Creation Diffusion Utilization*** 15.4, pp. 784–796 (cit. on p. 15).

Hendler, Jim and Andrew Hugill (2011). ‘The Syzygy Surfer : Creative Technology for the World Wide Web’. In: ***ACM WebSci 11***.

Hersovici, M et al. (1998). ‘The shark-search algorithm. An application: tailored Web site mapping’. In: ***Computer Networks and ISDN Systems*** 30.1-7, pp. 317–326 (cit. on p. 15).

Hotho, Andreas et al. (2006). ‘Information retrieval in folksonomies: Search and ranking’. In: ***The Semantic Web: Research and Applications, volume 4011 of LNAI***. Springer, pp. 411–426 (cit. on p. 15).

Hugill, Andrew (2012). ***’Pataphysics: A Useless Guide***. Cambridge, Massachusetts: MIT Press.

Jarry, Alfred (1996). ***Exploits and Opinions of Dr Faustroll, Pataphysician***. Cambridge, MA: Exact Change (cit. on pp. 5, 21).

– (2006). ***Collected Works II - Three Early Novels***. Ed. by Alastair Brotchie and Paul Edwards. London: Atlas Press.

Jeh, Glen and Jennifer Widom (2002). ‘SimRank: A Measure of Structural Context Similarity’. In: ***In KDD***, pp. 538–543 (cit. on p. 15).

Jordanous, Anna Katerina (2011). ‘Evaluating Evaluation : Assessing Progress in Computational Creativity Research’. In: ***Proceedings of the Second International Conference on Computational Creativity***.

– (2012). ‘Evaluating Computational Creativity: A Standardised Procedure for Evaluating Creative Systems and its Application’. PhD thesis. University of Sussex.

Jordanous, Anna Katerina and Bill Keller (2012). ‘Weaving creativity into the Semantic Web: a language-processing approach’. In: ***Proceedings of the 3rd International Conference on Computational Creativity***, pp. 216–220.

Jurafsky, Daniel and James H Martin (2009). ***Speech and Language Processing***. London: Pearson Education (cit. on pp. 16, 19, 20, 22).

Kamps, Jaap, Rianne Kaptein and Marijn Koolen (2010). ***Using Anchor Text , Spam Filtering and Wikipedia for Web Search and Entity Ranking***. Tech. rep. ? (Cit. on p. 15).

Kleinberg, Jon M (1999). 'Authoritative sources in a hyperlinked environment'. In: ***journal of the ACM*** 46.5, pp. 604–632 (cit. on p. 14).

Kleinberg, Jon M et al. (1999). 'The Web as a graph : measurements, models and methods'. In: ***Computer*** (cit. on p. 14).

Luke, Saint (2005). ***The Gospel According to St. Luke***. Ebible.org (cit. on p. 5).

Luo, Fang-fang, Guo-long Chen and Wen-zhong Guo (2005). 'An Improved 'Fish-search' Algorithm for Information Retrieval'. In: ***2005 International Conference on Natural Language Processing and Knowledge Engineering***, pp. 523–528 (cit. on p. 15).

Macdonald, Craig (2009). 'The Voting Model for People Search'. In: ***Philosophy*** (cit. on p. 11).

Manning, Christopher, Prabhakar Raghavan and Hinrich Schuetze (2009). ***Introduction to Information Retrieval***. Cambridge UP (cit. on p. 17).

Marchionini, Gary (2006). 'From finding to understanding'. In: ***Communications of the ACM*** 49.4, pp. 41–46 (cit. on p. 12).

Marchionini, Gary and Ben Shneiderman (1988). 'Finding facts vs. browsing knowledge in hypertext systems'. In: ***Computer*** 21.1, pp. 70–80 (cit. on pp. 5, 11).

Marcus, Mitchell P, Beatrice Santorini and Mary Ann Marcinkiewicz (1993). 'Building a Large Annotated Corpus of English: The Penn Treebank'. In: ***Computational Linguistics*** 19.2 (cit. on p. 21).

Mayer, Richard E (1999). 'Fifty Years of Creativity Research'. In: ***Handbook of Creativity***. Ed. by Robert J Sternberg. New York: Cambridge University Press. Chap. 22, pp. 449–460.

Mayhaymate (2012). ***File:PageRank-hi-res.png***. URL: `https://commons.wikimedia.org/wiki/File:PageRank-hi-res.png` (visited on 18/10/2016) (cit. on p. 14).

Michelsen, Maria Hagsten and Ole Bjorn Michelsen (2016). ***Regex Crossword***. URL: `http://regexcrossword.com/` (visited on 19/10/2016) (cit. on p. 18).

Microsoft (2012). ***Bing Fact Sheet*** (cit. on p. 5).

Miller, George A. (1995). 'WordNet: a lexical database for English'. In: ***Communications of the ACM*** 38.11, pp. 39–41 (cit. on p. 17).

Miyamoto, Sadaaki (1988). ***Information Retrieval based on Fuzzy Associations*** (cit. on p. 11).

– (2010). ***Fuzzy Sets in Information Retrieval and Cluster Analysis (Theory and Decision Library D)***. Springer, p. 276 (cit. on p. 11).

Miyamoto, Sadaaki and K Nakayama (1986). 'Fuzzy Information Retrieval Based on a Fuzzy Pseudothesaurus'. In: ***IEEE Transactions on Systems, Man and Cybernetics*** 16.2, pp. 278–282 (cit. on p. 11).

Nick, Z.Z. and P. Themis (2001). 'Web Search Using a Genetic Algorithm'. In: ***IEEE Internet Computing*** 5.2, pp. 18–26 (cit. on p. 15).

Nicole (2010). ***The 10 Most Incredible Google Bombs*** (cit. on p. 13).

Pease, Alison and Simon Colton (2011). 'On impact and evaluation in Computational Creativity : A discussion of the Turing Test and an alternative proposal'. In: ***Proceedings of the AISB***.

Pease, Alison, Simon Colton et al. (2013). 'A Discussion on Serendipity in Creative Systems'. In: ***Proceedings of the 4th International Conference on Computational Creativity***. Vol. 1000. Sydney, Australia: University of Sydney, pp. 64–71.

Pease, Alison, Daniel Winterstein and Simon Colton (2001). 'Evaluating Machine Creativity'. In: ***Proceedings of ICCBR Workshop on Approaches to Creativity***, pp. 129–137.

Piffer, Davide (2012). 'Can creativity be measured? An attempt to clarify the notion of creativity and general directions for future research'. In: ***Thinking Skills and Creativity*** 7.3, pp. 258–264.

Polya, George (1957). ***How To Solve It***. 2nd. Princeton, New Jersey: Princeton University Press (cit. on p. 11).

Project, NLTK (2016). ***Natural Language Toolkit***. URL: http://www.nltk.org/ (visited on 18/10/2016) (cit. on p. 16).

Ritchie, Graeme (2001). 'Assessing creativity'. In: ***AISB '01 Symposium on Artificial Intelligence and Creativity in Arts and Science***. Proceedings of the AISB'01 Symposium on Artificial Intelligence, Creativity in Arts and Science, pp. 3–11.

– (2007). 'Some Empirical Criteria for Attributing Creativity to a Computer Program'. In: ***Minds and Machines*** 17.1, pp. 67–99.

– (2012). 'A closer look at creativity as search'. In: ***International Conference on Computational Creativity***, pp. 41–48.

Schmidhuber, Juergen (2006). ***New millennium AI and the Convergence of history***.

Schuetze, Hinrich (1998). 'Automatic Word Sense Discrimination'. In: ***Computational Linguistics*** (cit. on p. 11).

Schuetze, Hinrich and Jan Pedersen (1995). ***Information Retrieval Based on Word Senses*** (cit. on p. 11).

Shu, Bo and Subhash Kak (1999). 'A neural network-based intelligent metasearch engine'. In: ***Information Sciences*** 120 (cit. on p. 15).

Srinivasan, P (2001). 'Vocabulary mining for information retrieval: rough sets and fuzzy sets'. In: ***Information Processing and Management*** 37.1, pp. 15–38 (cit. on p. 11).

Sutcliffe, Alistrair and Mark Ennis (1998). 'Towards a cognitive theory of information retrieval'. In: ***Interacting with Computers*** 10, pp. 321–351 (cit. on p. 11).

Taye, Mohammad Mustafa (2009). 'Ontology Alignment Mechanisms for Improving Web-based Searching'. PhD thesis. De Montort University (cit. on p. 15).

Turing, Alan (2009). 'Computing Machinery and Intelligence'. In: ***Parsing the Turing Test***. Ed. by Robert Epstein, Gary Roberts and Grace Beber. Springer. Chap. 3, pp. 23–66.

University, Princeton (2010). ***What is WordNet?*** URL: http://wordnet.princeton.edu (visited on 20/10/2016) (cit. on pp. 16, 17).

Varshney, Lav R et al. (2013). 'Cognition as a Part of Computational Creativity'. In: ***12th International IEEE Conference on Cognitive Informatics and Cognitive Computing***. New York City, USA, pp. 36–43.

Ventura, Dan (2008). 'A Reductio Ad Absurdum Experiment in Sufficiency for Evaluating (Computational) Creative Systems'. In: ***5th International Joint Workshop on Computational Creativty***. Madrid, Spain.

Verne, Jules (2010). ***A Journey to the Interior of the Earth***. Project Gutenberg (cit. on p. 6).

Vries, Erica de (1993). 'Browsing vs Searching'. In: ***OCTO report 93/02*** (cit. on p. 12).

Wickson, F., A.L. Carew and A.W. Russell (2006). 'Transdisciplinary research: characteristics, quandaries and quality'. In: ***Futures*** 38.9, pp. 1046–1059.

Widyantoro, D.H. and J. Yen (2001). 'A fuzzy ontology-based abstract search engine and its user studies'. In: ***10th IEEE International Conference on Fuzzy Systems*** 2, pp. 1291–1294 (cit. on pp. 11, 15).

Wiggins, Geraint A (2006). 'A preliminary framework for description, analysis and comparison of creative systems'. In: ***Knowledge Based Systems*** 19.7, pp. 449–458.

KTHXBYE

Go to TOC