Informatics Research Review

Mark van Rossum
University of Edinburgh

These slides are online.

(slides adapted from earlier versions by Alan Bundy, Alex Lascarides, Victor Lavrenko, Stratis Viglas)
Two components of MSc

• Taught component (100 credits)
  – lectures, tutorials, coursework, exams
  – learn established techniques that work

• Research component (80 credits)
  – do something that’s never been done before
    – study a new problem, develop a new method, etc.
    – probably the most exciting (and hardest) part of MSc
  – culminates in you writing a dissertation (~50 pages)
  – two courses prepare you:
    – IRR: literature review in area of interest
    – IRP: write a detailed plan for your MSc project
MSc project timeline

• Semester 1 (IRR)
  – learn about a relevant area: explore research papers
  – write a 3000-word summary of what you learned

• January:
  – faculty supervisors propose project topics
  – Or, propose your own
  – talk to supervisors, pick set of topics

• Semester 2 (IRP):
  – write a detailed research plan for what you’re going to do

• Summer (provided you pass 120 credits with 50% average)
  – work on your project (build, test, analyse results)
  – write a dissertation
IRR: what is it?

• **Compulsory: you have to show up.**

• **Survey** of research in targeted area

• **May be** forerunner to summer project
  
  • the literature survey can be *unrelated* to your exact summer *project*
  
  • **choice** of project done *after IRR*

• Delivered by a *teaching assistant* (TA)
  
  • normally a PhD student
  
  • knowledgeable about your *specialism*

• Approximately *3000 words (~6 full pages)*
  
  • you can write more, but one would not expect less
Purpose of IRR

1. Learn skills of *research reading*

2. Learn skills of *research writing*

3. *Refine* choice of research area

4. Learn *background* to project area
In a nutshell

• *Identify* relevant *papers: (see below)*

• *Keep notes* on each paper

• Attend related seminars: [http://www.inf.ed.ac.uk/events/seminars](http://www.inf.ed.ac.uk/events/seminars)

• Weave these into a *story*

• *Write* your report

• *Deadline: early in Semester 2 (typically late January)*
IRR / IRP structure

- Sign up to a group based on preferred interest
- IRR topic can differ from thesis topic, IRP and specialism
- [http://doodle.com/poll/XXXXX](http://doodle.com/poll/XXXXX) (see email) Please Name+S-number.

Groups are created based on Doodle poll

Groups will meet every week (starting in week 3(tbc))

- guide you through the stages of writing a review
- discuss progress, answer questions, provide feedback
- your tutor will email to arrange meeting time / place
- attendance mandatory – will affect mark
MSc Specialism Areas

- Agents, Knowledge Management, Representation & Reasoning (might split off agents)
- Analytical & Scientific Databases
- Bioinformatics, Systems & Synthetic Biology
- Cognitive Science
- Computer Systems, Software Engineering, High-Performance Computing
- Cyber Security and Privacy
- Intelligent Robotics
- Learning from Data / Data Science
- Natural Language Processing
- Neural Computation & Neuroinformatics
- Theoretical Computer Science
Suggested IRR tutor group timeline

Exact Location, time, and contents determined by tutor

- What is a literature review
- How to search for papers
- How to read papers effectively
- Write an introduction of your report
- Scientific writing
- Using Turnitin, reviewing drafts
- First draft of report due
- Feedback on first draft

Submit your report via Turnitin + hardcopy to ITO
Structure of review

- **Introduction**: identify and motivate topic

- **Main body**
  - **Summarize** each piece of work
  - Give critical **analysis**
    - Big contrast to taught courses: Research is “work in progress”
    - Not everything you read is correct or as important as the authors say.
  - **Compare** and **contrast**

- **Conclusion**
  - What is **state** of the field?
  - Where **next**?

- **Bibliography**: list all (and only) papers cited
How to identify papers to read

• **Select** interesting seed papers from:
  
  • https://www.wiki.ed.ac.uk/display/irrirpwiki/IRRSeedPapers
  
  • provided by project supervisors for each specialization

• **Follow-up** the *citations* in the papers you read (reference list)

• **See** who *cited* the paper (*easy with Google Scholar*)

• Library and online resources
  
  • Citeseer and ISI Web of Knowledge
  
  • Google Scholar
  
  • Library Online [http://www.lib.ed.ac.uk/resources](http://www.lib.ed.ac.uk/resources)
Active attitude: Always have questions in mind when reading (or listening to a talk)

- What are the *aims* and *objectives* of the work?
- What was *achieved*?
- What *claims* are being made?
- Is the *supporting evidence* convincing?
- What would I do to extent work / disprove claims.
Types of scientific papers

• A new way of doing something (algorithm, analysis, technique)
• Discovery of something
• Report a correlation/relation “Smokers have higher IQ”
• Linking previously unrelated findings/methods; application to new area.
• Over-arching description/understanding
• Review of the field
• ....
Typical claims in Informatics

X is better than Y on task Z along some dimension W

• What kind of things are X and Y?
  • system?
  • technique?
  • parameter?
• What is task Z?
• What is the dimension W?
  • behaviour, coverage, efficiency, usability, dependability, maintainability

For keyword-based searches in medical databases, Pseudo-Relevance Feedback will provide better search results than Topic Modeling as measured by mean average precision of the ranked list.
Hypotheses in Informatics

• *Hypotheses/claims* often not stated

  • *except* in *theoretical* work

  • leads to confusion and misunderstanding

• If *claim not clear* then this should be *criticised*

  • same if claim is *strong* and is *not proven*

• *Evidence* may be *theoretical* or *experimental*

• *Objective* may be to *identify a hypothesis* for subsequent evaluation
Reading to different depths

• *Some work* is *central* to your concerns, *some less so*

• Need to *vary reading depth*
  
  • some need only *skim*
  
  • some read *in depth*
  
  • some *in between*

• Re-read a paper at a later stage. Your opinion will probably change.

• Could be *20+ papers in total*, but only *3 or 4 in depth*
  
  • but you need to *cite everything*
How to skim

• *Read* title, abstract, introduction, conclusion, bibliography, *key sections*

• *Identify* main *contribution* of paper

• How does it *relate* to other work?

• *Identify* key *questions to be addressed* and hunt for answers
How to read in depth

• Make *several passes* over the paper

  • start by skimming

  • then read in increasing detail

• *Apply techniques* to your *own examples (thought experiment)*

• *Try explaining* the ideas to a friend, or even a virtual friend.
Telling a story

• Literature survey is part of *motivation*

• How did this field *develop*?

• How did it *start*?

• What are the *rival approaches*?

• How do pieces of work *relate*?

• Where are we *now*?

• What remains *to be done*?

• What are the *hot topics*?
Pacing Yourself

• Work out *timetable* for reading/writing

• Leave *plenty of time for feedback* and correction

• Read at a *steady pace*

• Keep notes

• *Write as you go*
Technical writing

• Your audience: a fellow student: knows some basics, but has not read the papers that you have read.

• Like software writing: there are guidelines, but no recipes for good writing. It starts with thinking (away from the computer).

• The reader of a scientific paper will want to *know exactly what is going on*
  • it’s *not* a *mystery novel*; there is no plot, only facts and (maybe) opinion
  • do *not try* to write *flamboyantly*; it *confuses and irritates* the reader
  • use *terminology*; it’s there for a reason

• *Strunk & White: The elements of style* (available online)

• [Examples of great writing: Richard Dawkins The Oxford Book of Modern Science Writing]
Example of what to avoid

“[...] to enable the advantageous employment of the inherent stratification of the memory infrastructure of modern processors and their interconnected means of communication [...]”

“[...] to better use the cache hierarchy and the system bus [...]”
Bibliography

• List all and only papers cited


• There are several styles and publication types

  • LaTeX/LyX supports several

• Provide the following:

  • author(s) name(s), paper title, journal/book/conference title, year, pagination, volume/number, editor(s), publisher
Assessment

• Your report will be marked by your tutor, you will receive feedback

• Mark is based on:

  • **Appropriate coverage**: did you hit all the important papers in the area?

  • **Understanding of sources**: are you just parroting back what you read?

  • **Critical evaluation and comparison**: beyond “A did X, B did Y”?

  • **Clarity of expression and presentation**: do your friends understand it?

  • **Attendance of tutorial group meetings**: discuss all absences with your tutor
Assessment

We are here to help and most students enjoy the course.

Reason for failure:

• Plagiarism

• Not showing up at tutorials

• Non-scientific topic: 'the new iPhone', 'History of Facebook'
Avoid plagiarism

• *Quotations* must be acknowledged
  
  • including close paraphrase

  • use quote marks and cite source
    ...Smith (2009) argued that “the Level 2 cache systems are the core to fast database systems in future HPC”....

• *Do not* copy-paste-edit from online sources

• Read School guide on plagiarism


• Plagiarism carries *serious penalties*. Fail on the course is the least severe...
In summary, Linked Data is simply about using the Web to create typed links between data from different sources. These may be as diverse as databases maintained by two organizations in different geographical locations, or simply heterogeneous systems within one organization that, historically, have not easily interoperated at the data level.

Technically, Linked Data refers to data published on the Web in such a way that it is machine-readable, its meaning is explicitly defined, it is linked to other external data sets, and can in turn be linked to from external data sets.

While the primary units of the hypertext Web are HTML documents connected by untyped hyperlinks, Linked Data relies on documents containing data in RDF. However, rather than simply connecting these documents, Linked Data uses RDF to make typed statements that link arbitrary things in the world. The result, which we will refer to as the Web of Data, may more accurately be described as a web of things in the world, described by data on the Web.[6]

Berners-Lee outlined a set of 'rules' for publishing data on the Web in a way that all published data becomes part of a single global data space:

1. Use URIs as names for things
2. Use HTTP URIs so that people can look up those names
3. When someone looks up a URI, provide useful information, using the standards (RDF, SPARQL)
4. Include links to other URIs, so that they can discover more things.

These have become known as the 'Linked Data principles', and provide a basic recipe for publishing and connecting data using the infrastructure of the Web while adhering to its architecture and standards.

The most visible example of adoption and application of the Linked Data principles has been the Linking Open Data project, a grassroots community effort founded in January 2007 and supported by the W3C Semantic Web Education and Outreach Group. The original and ongoing aim of the project is to bootstrap the Web of Data by identifying existing data sets that are available under open licenses, converting these to RDF according to the Linked Data principles, and publishing them on the Web.
Histogram equalization is one of the most commonly used methods for contrast enhancement. It refers to the modification of the image through some pixel mapping such that the histogram of the processed image is more uniform than that of the original image (Arici, 2009). Since it deals with direct manipulation of the pixel intensity values, thus it falls within the category of image enhancement in spatial domain. Histogram equalization method can be achieved by using the cumulative density function of the input image to derive uniform distributed histogram of the enhanced image (Arici, 2009). Ibrahim & Kong (2007) introduced Brightness Preserving Dynamic Histogram Equalization (BPDHE) method.

1 Brightness Preserving Dynamic Histogram Equalization (BPDHE)

BPDHE is a histogram equalization method, which produces the output image with the mean intensity almost equivalent to the mean intensity of the input image (Ibrahim & Kong, 2007). The steps for BPDHE are as follows:

Method 1: Smooth the histogram with a Gaussian filter

In Figure 2, we can see that the histogram of the digital image is fluctuated and the probability for some intensity values are missing (i.e. intensity values from 0-39 as well as 111-256). As stated by Ibrahim & Kong (2007), it is difficult to detect the local maximums of the histogram without smoothing the histogram. Therefore, the first step is to fill up the disappeared intensity values by using linear interpolation. Linear interpolation is a first degree method that passes a straight line through every two consecutive points of the input signal. If the two known points are given by the coordinates \((x_0, y_0)\) and \((x_1, y_1)\), then the linear interpolant is the straight line between these two points. For a value \(x\) in the interval, the value \(y\) along the straight line is given by the equation:

\[
\frac{y - y_0}{x - x_0} = \frac{y_1 - y_0}{x_1 - x_0}
\]

Solving the equation for \(y\) gives

\[
y = y_0 + \left( x - x_0 \right) \frac{y_1 - y_0}{x_1 - x_0}
\]

Another way to fill up the disappeared intensity values is through the fractal interpolation method as introduced by Kobes & Letkeman (2001). In fractal interpolation, the video frame is encoded into fractal codes via fractal compression, and subsequently decompressed at a higher resolution. For the classic linear Fractal Interpolation Function (FIF), the Iterated Function System (IFS) is used to construct a FIF for a given set of
8.2.2.1 Methods

The goal of CRM model is to learn the joint distribution $P(r, w)$ over regions $r$ of an image and the words $w$ in its annotation (Lavrenko, Manmatha, & Jeon, 2003). As stated in Lavrenko, Manmatha and Jeon (2003), $T$ is the training set of annotated images, $J$ is an element of $r_A = \{r_1, ..., r_{nA}\}$ is the regions of some image $A$, that is in the testing set of unannotated images and $w_B = \{w_1, ..., w_{nB}\}$ is some arbitrary sequence of words, we can compute a joint probability, $P(r_A, w_B)$ by:

$$P(r_A, w_B) = \sum_{J \in T} \prod_{b=1}^{n_B} P_v(w_b \mid J) \prod_{a=1}^{n_A} \int_{\mathbb{R}^k} P_R(r_a \mid g_a) P_G(g_a \mid J) dg_a$$

The estimation parameters are:

1. $P_T(J)$ which is a uniform prior, is the probability of selecting underlying model of image $J$ to generate some new observation $r, w$ (Lavrenko, Manmatha, & Jeon, 2003).

$$P_T(J) = \frac{1}{N_T}$$

where $N_T$ is the size of the training set.

2. $P_R(r \mid g)$ is a global probability distribution used to map generator vectors, $g \in \mathbb{R}^k$ to actual image regions $r \in R$ (Lavrenko, Manmatha, & Jeon, 2003).

$$P_R(r \mid g) = \begin{cases} \frac{1}{N_g} & \text{if } G(r) = g \\ 0 & \text{otherwise} \end{cases}$$

where $N_g$ is the number of all regions $r'$ in $R$ such that $G(r') = g$.

3. $P_G(\cdot \mid J)$ is a non-parametric kernel-based density function used to generate the feature vectors $\{g_1, ..., g_n\}$, that are later mapped to image regions $r_J$ according to $P_R$ (Lavrenko, Manmatha, & Jeon, 2003).

$$P_G(g \mid J) = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{\sqrt{2\pi k_i^2}} \exp\left\{-(g - G(r_i))^T \sum_i^{-1} (g - G(r_i))\right\}$$

where $r_J = \{r_1, ..., r_n\}$ is the set of regions for image $J$, $\sum$ is the feature covariance matrix with equation $\sum = \beta I$, where $I$ is the identity matrix and $\beta$ plays the role of kernel bandwidth that determine the smoothness of $P_G$ around support point $G(r_i)$.

4. $P_v(\cdot \mid J)$ is the multinomial distribution that is assumed to have generated the annotation $w_J$ of image $J \in T$ (Lavrenko, Manmatha, & Jeon, 2003).
An email....

“I am addressing you this mail on the subject of the charges of plagiarism I have recently faced for my IRR report. I have had a meeting with the Academic Misconduct Panel last week, where I explained how the situation had come about, why I committed those errors and some circumstances I feel contributed to the lack of understanding of my part. I just got back the decision of the Panel. I was informed that they feel it is a major case of plagiarism and, therefore they give me a penalty of 30% of the total available mark. Given the fact that my note was borderline this penalty means I fail IRR and that I cannot present a master thesis.”
“I am addressing you this mail on the subject of the charges of plagiarism I have recently faced for my IRR report. I have had a meeting with the Academic Misconduct Panel last week, where I explained how the situation had come about, why I committed those errors and some circumstances I feel contributed to the lack of understanding of my part. I just got back the decision of the Panel. I was informed that they feel it is a major case of plagiarism and, therefore they give me a penalty of 30% of the total available mark. Given the fact that my note was borderline this penalty means I fail IRR and that I cannot present a master thesis.”
Summary

• Make thoughtful and thorough *search for sources*
  • Papers and seminars
• Study each in *appropriate depth*
  • Keep notes
• Critically *evaluate and compare*
  • Weave into story
• *Write* ~3000 word report
  • Reflect story and relevance in report
• *Leave time* for feedback by your tutor and correction
To do now

• Fill in Doodle poll (name + S-number)
• You can start browsing Seed papers.
More information

• Wiki: www.wiki.ed.ac.uk/display/irrirpwiki/IRR+and+IRP+Wiki

• MSc project guide

• If you have questions
  • ask the TA of your group
  • make an appointment to see me