How to give a good research talk

Andreas Zeller
Goals of the Seminar

• Find your way into scientific challenges
• Structure and present scientific material
• Train your social and communication skills
The Purpose of your Talk
The Purpose of your Talk
The Purpose of your Talk

• Make the audience read your paper 
  (and talk about it)
• Give them an intuitive feel for your idea
• Engage, excite, provoke them
• Make them glad they came
Preparation

• Check the material
• Identify central topics and claims
• Outline the talk
• Make a detailed sketch
Ask Yourself

• Do the claims hold?
• Are the examples illustrative?
• Can I do better in presenting?
• What are the central claims, anyway?
• And how are they supported?
Ask Yourself

• If someone remembers *one thing* from my research talk, what should it be?
The Perfect Talk

- I'd like to start with a diagram.
- It's a bunch of shapes connected by lines.
- Now I will say some impressive words.
- Synchronized incremental digital integrated dynamic e-commerce space.
- Any questions?
- May I have a copy of your presentation?
- The results of my experiment are disturbing.
Your Audience

• Have read all your earlier papers

• Thoroughly understand Computational Complexity of Bio-inspired Computation in Combinatorial Optimization

• Are eagerly awaiting your latest and greatest

• Are fresh, alert, and ready for action

• Have never heard of you

• Have heard of it, but wish they had not

• Could not care less

• Just came back from lunch and are ready for a nap
Your Audience
Organizing Your Talk

- Motivation
- Solution (including failures)
- Results
- Conclusion
Motivation

- Present the general topic
  *A village in the woods*

- Show a concrete problem  
  (and make it the audience’s problem)  
  *Wicked dragon attacks the peasants*

- Show that the state of the art is not enough  
  *Peasants’ forks can not pierce dragon armor*
Solution + Results

• Show new approach and its advantages
  *Hero comes with vorpal blade and fights dragon*

• Show how approach solves concrete problem
  *Vorpal blade goes snicker-snick; dragon is slayed*

• Does the approach generalize?
  *Would this work for other dragons, too? Why?*
Examples: Your main Weapon

- Motivate work
- Convey basic intuition
- Illustrate idea in action
- Use *examples* first, *generalize* afterwards
Outline

• Tell a story
• Make slides invisible
• Use examples, lots of examples
• Connect to the audience
• Hope for questions and feedback
Outlines

• Don’t use talk outlines at the beginning
• Don’t use talk outlines in between
• Actually, don’t use talk outlines at all
• Better: Use a diagram after 5 minutes
• Think of this diagram as a memorizable image
How do we know a program does what it claims to do? After clustering applications (1), we analyze their natural language description (2) to form clusters of related apps (3). For each cluster, we identify the APIs (4) that are used, grouped by related permits (5). For unusual APIs, such as querying GPS location, we classify the app as anomalous (6). In this way, we can identify potential malware patterns and prevent unauthorized access to confidential data.

Figure 1: Detecting applications with unadvertised behavior.
Slide Contents

• Concentrate on the bare necessities (e.g. at most 5 bullets per slide)

• Do not present full sentences on a slide, because these are far too long and hard to read; also, they may tempt you in reading them loud.
Death by Powerpoint

LIFE AFTER DEATH
BY POWERPOINT

with DON McMILLAN
Stemming

looking for a restaurant, a bar, a pub or just to have fun in London? search no more! this application has all the information you need:

• you can search for every type of food you want: french, british, chinese, indian etc.
• you can use it if you are in a car, on a bicycle or walking
• you can view all objectives on the map
• you can search objectives
• you can view objectives near you
• you can view directions (visual route, distance and duration)
• you can use it with street view
• you can use it with navigation

keywords: London, restaurants, bars, pubs, food, breakfast, lunch, dinner, meal, eat, supper, street view, navigation
Stemming

looking for a restaurant, a bar, a pub or just to have fun in London? search no more! this application has all the information you need:

• you can search for every type of food you want: french, british, chinese, indian etc.
• you can use it if you are in a car, on a bicycle or walking
• you can view all objectives on the map
• you can search objectives
• you can view objectives near you
• you can view directions (visual route, distance and duration)
• you can use it with street view
• you can use it with navigation
keywords: london, restaurants, bars, pubs, food, breakfast, lunch, dinner, meal, eat, supper, street view, navigation
Stemming

look london restaur search bar pub just applic fun
inform can search need everi type food want french
british chines indian etc car bicycl walk
can us can view object map visual rout
can search object search can view distanc
durat can view direct object near
can us street view can us navig
keyword london restaur bar pub food view
breakfast lunch dinner meal eat supper street navig
Make Slides Invisible

- Focus on *clarity*
- Avoid all that distracts from the message
- Slides should *support* your (spoken) word
- Always prefer diagrams over text
- Avoid bullet lists (like this one)
“Travel” Cluster
\[ f_{h, \varepsilon}(x, y) = \varepsilon E_{x, y} \int_0^{t_\varepsilon} L_{x, y_\varepsilon(\varepsilon u)} \varphi(x) \, du \]
\[ = h \int L_{x, z} \varphi(x) \rho_x(dz) \]
\[ + h \left[ \frac{1}{t_\varepsilon} \left( E_y \int_0^{t_\varepsilon} L_{x, y(x(s))} \varphi(x) \, ds - t_\varepsilon \int L_{x, z} \varphi(x) \rho_x(dz) \right) \right. \]
\[ + \frac{1}{t_\varepsilon} \left( E_y \int_0^{t_\varepsilon} L_{x, y(x(s))} \varphi(x) \, ds - E_{x, y} \int_0^{t_\varepsilon} L_{x, y_\varepsilon(\varepsilon s)} \varphi(x) \, ds \right) \right] \]
\[ = h \hat{L}_x \varphi(x) + h \theta_\varepsilon(x, y) \]
Formal Background

Concrete state $\nu \in V$ with $\nu = (x_1, x_2, \ldots, x_n)$

$x_i$ — Return value of an inspector

Trace $t = [(\nu_1, m_1, \nu_1'), (\nu_2, m_2, \nu_2'), \ldots]$ with $\nu_i \in V$ and $m_i$ — name of a mutator

State abstraction $\text{abs}: V \to S$

Model with transitions $s \xrightarrow{m} s'$ and states $s, s' \in S$

Transition condition $s \xrightarrow{m} s'$ with $s, s' \in S$ iff

$\exists (\nu, m, \nu') \in t \cdot \text{abs}(\nu) = s \land \text{abs}(\nu') = s'$
Maths

• Avoid maths.
  • Formulae are for papers, not slides
  • Few people can read + understand complex formulae in 30 seconds
• Demonstrate that the formal foundation can be presented on demand
Examples

• Examples are more important than maths

• Have one example throughout your talk to illustrate the key idea

• Use additional examples for specifics

• Your audience will get excited by the example – and read your paper for the full foundations
int cgi_decode(char *encoded, char *decoded)
{
    char *eptr = encoded;
    char *dptr = decoded;
    int ok = 0;
    char c;
    c = *eptr;
    if (c == '+') {
        *dptr = ' ';
    } else if (c == '%') {
        *dptr = *eptr;
    } else {
        *dptr = *eptr;
    }
    while (*eptr) {
        c = *eptr;
        if (c == '+') {
            *dptr = ' ';
        } else if (c == '%') {
            *dptr = *eptr;
        } else {
            *dptr = *eptr;
        }
        int digit_high = Hex_Values[*++eptr];
        int digit_low = Hex_Values[*++eptr];
        if (digit_high == -1 || digit_low == -1) {
            ok = 1;
        } else {
            *dptr = 16 * digit_high + digit_low;
        }
        ok = 0;
        ++dptr;
        ++eptr;
    }
    *dptr = '\0';
    return ok;
}
```c
int cgi_decode(char *encoded, char *decoded)
{
    char *eptr = encoded;
    char *dptr = decoded;
    int ok = 0;

    while (*eptr) {
        char c;
        c = *eptr;
        if (c == '+') {
            *dptr = ' ';
        } else if (c == '%') {
            *dptr = *eptr;
        } else {
            *dptr = *eptr;
        }
        int digit_high = Hex_Values[*(++eptr)];
        int digit_low = Hex_Values[*(++eptr)];
        if (digit_high == -1 || digit_low == -1) {
            ok = 1;
        } else {
            *dptr = 16 * digit_high + digit_low;
            ok = 1;
        }
        ++dptr;
        ++eptr;
    }
    *dptr = '0';
    return ok;
}
```
```c
int cgi_decode(char *encoded, char *decoded)
{
    char *eptr = encoded;
    char *dptr = decoded;
    int ok = 0;

    while (*eptr) {
        char c;
        c = *eptr;
        if (c == '+') {
            *dptr = ' ';
        } else if (c == '%') {
            *dptr = *eptr;
        } else {
            *dptr = *eptr;
            int digit_high = Hex_Values[*(++eptr)];
            int digit_low = Hex_Values[*(++eptr)];
            if (digit_high == -1 || digit_low == -1) {
                ok = 1;
            } else {
                *dptr = 16 * digit_high + digit_low;
            }
        }
        ++dptr;
        ++eptr;
    }

    *dptr = "0";
    return ok;
}
```
```c
int cgi_decode(char *encoded, char *decoded)
{
    char *eptr = encoded;
    char *dptr = decoded;
    int ok = 0;
    char c;

    while (*eptr) {
        c = *eptr;
        if (c == '+') {
            *dptr = ' ';
        } else if (c == '%') {
            *dptr = *eptr;
        } else {
            *dptr = *eptr;
        }
        int digit_high = Hex_Values[*(++eptr)];
        int digit_low = Hex_Values[*(++eptr)];
        if (digit_high == -1 || digit_low == -1) {
            ok = 1;
        } else {
            *dptr = 16 * digit_high + digit_low;
            ++dptr;
            ++eptr;
        }
    }

    *dptr = '0';
    return ok;
}
```
```c
int cgi_decode(char *encoded, char *decoded)
{
    char *eptr = encoded;
    char *dptr = decoded;
    int ok = 0;

    while (*eptr) {
        char c;
        c = *eptr;
        if (c == '+') {
            *dptr = ' ';
        } else if (c == '%') {
            *dptr = *eptr;
        } else {
            *dptr = *eptr;
        }
        int digit_high = Hex_Values[*(++eptr)];
        int digit_low = Hex_Values[*(++eptr)];
        if (digit_high == -1 || digit_low == -1) {
            ok = 1;
        } else {
            *dptr = 16 * digit_high + digit_low;
        }
        ++dptr;
        ++eptr;
    }

    *dptr = "0";
    return ok;
}
```
Diagrams

- Use simple, clear diagrams
- Convey exactly *one* message per diagram
### App Classification

With Clusters (our approach)

<table>
<thead>
<tr>
<th></th>
<th>Predicted as Malicious</th>
<th>Predicted as Benign</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Malicious Apps</strong></td>
<td>56 %</td>
<td>44 %</td>
</tr>
<tr>
<td><strong>Benign Apps</strong></td>
<td>16 %</td>
<td>84 %</td>
</tr>
</tbody>
</table>
Correct Classification
With Clusters (our approach)

Malicious Apps: 56%
Benign Apps: 84%
Visuals and Animation

- Visuals and animations are ok in *diagrams*
- Every other use should be well motivated
- Do not use them as decorations
- Do not use them as distractions
- Avoid overused graphic clichés
What's Wrong?

What people are saying about our new techniques

- Powerful
- Successful
- Rewarding
Death by Powerpoint
Strive for Simplicity

• Simple *messages* get across easier
• Simple *examples* fit on one slide
• Simple *slides* make the audience listen
• Simple *claims* tend to be general, too
• Simple = Hard!
The Talk

- Do not read your slides (from paper or slides)
- Speak slowly, loudly and clearly
- Speak personally (Use “I”, not “one”)
- Change your tone – and use pauses
The Jelly Factor

• Every presenter is nervous (and so am I)
  • Legs start shaking
  • Need for air
  • Brain goes into stand-by mode
• ... but nobody will notice, let alone worry
The Jelly Factor

Before the talk:

• Wash your hands
• Sit down
• Go through your slides
• Memorize the first sentences (no brain required)
Your Impression

Body language: 55%
Voice: 38%
Content: 7%
Connect to the Audience

• Tell a *story*

• Talk *directly* to the audience

• Ask *rhetorical questions*  
  (“What should the poor peasants do?”)

• Search *eye contact* to audience  
  (not to slides, not to professor)

• Convey your own *enthusiasm and excitement!*
Some Great Presenters
Lawrence Lessig
Concluding the Talk

• Refer to the beginning
  ...and they lived in peace henceforth

• Summarize
  ...and the key point is:

• Open issues
  ...but there are more dragons that loom in the dark

• Consequences
  If you ever see a dragon, ...
Checking App Behavior Against App Descriptions

Andreas Zeller
Saarland University, Saarbrücken, Germany
Joint work with Alessandra Gorla, Ilaria Tavecchia, and Florian Gross

http://www.st.cs.uni-saarland.de/chabada/

Chabada

1. App collection
2. Topics
3. Clusters

Internet Access-Location

Travel Cluster

Description

APIs used

London Restaurants

Key Findings

- Of the top 5 outliers per cluster, 26% show unadvertised (covert) behavior
- Typically ad frameworks (apploving, airpush)
- Several anomalies (UNO, WICKED, Yahoo! Mail...)
- Using OC-SVM as a classifier of APIs per cluster, we could flag 56% of novel malware as such
- Current work: Dynamic API usage, information flow, user authorization

http://www.st.cs.uni-saarland.de/chabada/
Any Questions?

• Good research raises lots of questions!

• Questions are great to connect to the audience and to direct and shape own work

• The worst embarrassment is to have no questions at all
Dealing with Hard Questions

• Repeat question (helpful for audience + gives time for preparing an answer)

• In doubt: “I don’t know, but I’ll look into it”

• Or: “Let’s just take this offline”

• Be respectful to the audience – no punching in the lecture room
The Purpose of your Talk

- Make the audience read your paper *and talk about it*
- Give them an *intuitive feel for your idea*
- Engage, excite, provoke them
- Make them glad they came

Make Slides Invisible

- Focus on *clarity*
- Avoid all that distracts from the message
- Slides should *support* your (spoken) word
- Always prefer diagrams over text
- Avoid bullet lists (like this one)

Summary

Examples

- Examples are more important than maths
- Have one example throughout your talk to illustrate the key idea
- Use additional examples for specifics
- Your audience will get excited by the example – and read your paper for the full foundations

Connect to the Audience

- Tell a *story*
- Talk *directly* to the audience
- Ask *rhetorical questions* (*“What should the poor peasants do?”*)
- Search *eye contact* to audience (not to slides, not to professor)
- Convey your own *enthusiasm and excitement!*
The Purpose of your Talk

- Make the audience read your paper (and talk about it)
- Give them an intuitive feel for your idea
- Engage, excite, provoke them
- Make them glad they came

Make Slides Invisible

- Focus on clarity
- Avoid all that distracts from the message
- Slides should support your (spoken) word
- Always prefer diagrams over text
- Avoid bullet lists (like this one)

Summary

Examples

- Examples are more important than maths
- Have one example throughout your talk to illustrate the key idea
- Use additional examples for specifics
- Your audience will get excited by the example – and read your paper for the full foundations

Connect to the Audience

- Tell a story
- Talk directly to the audience
- Ask rhetorical questions (“What should the poor peasants do?”)
- Search eye contact to audience (not to slides, not to professor)
- Convey your own enthusiasm and excitement!