CMSI/ELEC 601 GRADUATE SEMINAR

Dr. Stephanie E. August

A Guide to CMSI/ELEC 601 Graduate Seminar

This Guide is intended to help explain to Computer Science and Electrical Engineering students the expectations for successfully completing CMSI 601.

There are two major parts to your grade:

- Successful completion of the basic process of 601
- Presentation to the Electrical Engineering and Computer Science Faculty of a paper that meets the basic requirements

The basic process of 601 (see also the CMSI/ELEC 601 Graduate Seminar course overview handed out on the first day of class):

- Choose your advisor. You may choose any full-time faculty member from Electrical Engineering and Computer Science, and are not restricted to choosing your academic advisor. In the event that you cannot find a faculty member who is interested or believes in your project and with special permission from the Director of the Graduate Program in Electrical Engineering and Computer Science, you may ask an outside advisor to provide content advice *in addition to* a faculty advisor.
- Meet every other week with your advisor. It is advisable that you provide to your advisor all materials you will present during the semester you are taking 601 prior to presenting them in class and with sufficient time for you to receive back your advisor's comments.
- Ask your advisor to sign and date the front page of each document you turn in for the class, whether it is a PowerPoint or similar presentation, status report, bibliography, or paper draft. This assures the instructor that you have met with your advisor, and that your advisor is aware of your progress.
- Work with your advisor to reach an agreeable topic. See below. Complete this before the semester begins or by the second week of the semester at the absolute latest. The topic is to be accompanied by a list of the references you will use in your paper. You should have *at least* ten references. The majority must be from peer reviewed scholarly publications, as opposed to a trade magazine such as *PC AI* or *Database Programming and Design*. You should include a copy of the published abstract of each article, unless one does not exist.
- Write parts of your paper as you go and have your advisor review and approve them.
- Make sufficient progress at all milestones. Advisors will force students to take incompletes if they have not done sufficient work at each milestone. There is too much risk in a catch up game.
- Maintain a notebook containing your proposal, drafts, reports, course forms, references, notes on ideas, and feedback from the class. This will help you both remain organized and avoid going in circles with your ideas.

Acceptable Topic

The central point of any CMSI/ELEC 601 paper is:

The Problem

Your first challenge is to identify a problem that is small enough and focused enough to solve in one semester. Once you have the right problem, then the rest of the project is easy. You simply need to find a solution. Finding the problem is the difficult part, but it is not impossible.

Most topic areas eventually lead to a problem if carried far enough. Finding an appropriate topic requires being familiar with current work on the topic. Acceptable topics generally focus on problems that have not yet been addressed by others in the field.

We can illustrate with an example:

> *Sue comes to Dr. Toal with the idea that she would like to do something related to delivering training through the web or e-Learning. The basic area is fine, there are lots of good problems in there, but she isn't close to a topic. So, Dr. Toal would ask, "anything in particular within e-Learning delivery?" Sue then says that she finds most of the existing work boring. "Why is that?" Well, most of it isn't very relevant. "Ah, now we are getting somewhere." Her first cut at the problem, which is still probably too broad, is "How do you make e-Learning more relevant to users?" Now Sue must research what people are doing. Sue finds that there is a lot of research on smaller aspects of content called Reusable Content Objects (or RCO) and a lot of research on learning portals that should be able to find more relevant RCOs. However, Sue realizes that the current learning portals are mostly based on an individual's role alone and not on any other factors. However, some earlier research shows that training can be delivered in very fine-grained "chunks" based on knowledge nets. In Sue's literature search, she doesn't see anyone applying Knowledge Nets to portals. Sue is closing in on a topic. The problem has now become, "How can knowledge nets be used to find relevant content for users based on factors other than role?" This looks like a great problem.*
>
> *The solution is most likely obvious to Sue, since she is the one that realized there is some correlation between knowledge nets and portals. Sue is going to design a portal based on the knowledge net concept. She will show the basic design of database and will consider how actual job performance numbers could be factored into the system. Finally, she recognizes that this leaves her with a few additional problems that she won't be able to research, including how RCOs can be combined into larger chunks of knowledge more like courses.*
>
> *The last point is especially important. Sue has shown not only that she can identify one problem, but also that through her research she has been able to identify several more. Her work has come full circle.*

If you cannot conclude your talk with additional problems, then either you have defined your topic too narrowly or your topic is not very interesting. It is not necessary to completely solve your topic area.

Generally, we accept:

> *a problem and solution with sufficient technical depth to demonstrate your understanding of a complex technical problem related to computer science.*

Your advisor in many ways is the one who determines whether your problem and solution meet the additional criteria. The following are examples of topics that would not be acceptable:

- Solving world hunger – too broad and not technical enough, but definitely a problem
- Comparing approaches to data mining – a comparison is never sufficient. Comparisons must be turned into either a model that can be used to compare several things. For example, one of our faculty members once published a paper on a reference model for software engineering environments. Be careful about these models; there must be a need for the model. At the time he published the paper, there was no common understanding of comparisons between environments that made it easy to consider what the environments could achieve. Thus, a comparison of the effectiveness of data mining approaches over a particular type of problem set could be a perfectly good topic, if it has not been done before.
- Researching the latest optical fiber technology – two difficulties here. Simply researching something is never sufficient. Second, there is the question of whether optical fiber technology is closely related to computer science. It is essential for a computer science faculty member to agree to advise and review what you are specifically doing within your broad topic area.

The Paper and Presentation

The general format of your final project report should be:

1. *Abstract* – Two paragraphs. First one concludes with the problem statement. Second paragraph states your solution and what it contributes.
2. *Introduction*: Background information and overview of your topic. Includes your problem statement and a brief description of your solution. Ends with a basic description of the rest of the paper. If we think of a system as input+process+output, this section provides an example of both the *input to* and the *output of* the system.
3. *Related Work* – Survey of work already published and a discussion how it relates to your topic and reinforces your problem, and why each individual related work is not sufficient to solve the problem.
4. *Background* – Summary of the work on which your project is based and builds upon.
5. *Solution* – May be several chapters, but is the basic solution to the problem. If we think of a system as input+process+output, this section describes how you *transform* the input of the system into the output of the system.
6. *Conclusion and Future Research* – What is the contribution of your work to the field? And what else needs to be done?
7. *Annotated Bibliography* – The annotated reference list (see below).
8. *Related Documents* – An appendix containing any related documents, such as source code, diagrams, design, or specifications.

9. *Lessons Learned* – Turn in with final paper as a separate document.

   Describe what you learned during the course of this project. This discussion should be about 500 words in length. Your lessons learned section should at a minimum address the following questions: What would you do differently next time? What resources should you have used, or do you wish had been available? How many hours did you put in? How useful was the project in helping you learn the course material. What would have facilitated completing the project?

   This section is not optional!

You will receive a more complete document formatting guide on the first day of class.

Style of Writing

You are learning to write a scholarly, professional paper. We expect *all* drafts and final papers produced in 601 to have a style similar to that found in the scholarly articles that form the basis of the *Related Work* section of your final paper. This applies to quality of work, style of writing, and format. Study your articles with this in mind.

Annotated Bibliography

The *annotated bibliography* is a list of the papers you will be citing in your paper and a summary of each. Remember to cite relevant sources in the body of your paper! The style recommended by the American Psychological Association (APA) is a good choice. See the LMU Style Manual 2000 for details. Regardless of the style you select, be consistent, be complete, and include page numbers wherever possible. References should be cited using the author's surname and the year of publication. For example, the paper:

August, Stephanie E., and Dolan, Charles P. (1992) Hughes Research Laboratories: Trainable Text Skimmer for MUC-4, System Description. *Proceedings of the Fourth Message Understanding Conference (MUC-4).* Mc Lean, Virginia, 16-18 June.

would be cited as (August and Dolan, 1992). Each reference should be accompanied by a two to three sentence description of the reference. Ideally, the annotation will summarize the article, state the main issue(s) raised in the article, and show how the article is related to other work you have read. The annotated bibliography entry for the publication above would be:

August, Stephanie E., and Dolan, Charles P. (1992). Hughes Research Laboratories: Trainable Text Skimmer for MUC-4, System Description. Proceedings of the Fourth Message Understanding Conference (MUC-4). Mc Lean, Virginia, 16-18 June, 1992.

The Trainable Text Skimmer (TTS) is a statistical system which uses a set of Bayesian classifiers to extract data from free text input and to select and complete the slots of an appropriate event template. First, a set of frequently used phrases is identified in a training corpus. Next, a text is skimmed to determine which category of event template it most closely matches. Finally, relevant data is extracted from the text and used to complete the template. TTS is effective for the majority of slots, but it is thought that performance cannot be improved significantly without including language- or domain-specific processing, of which it currently contains very little.

Here are the key points to identify for each reference read. They provide a useful perspective overall for any kind of reading or research:

1. What was the work's contribution to the field?
2. What did the work leave undone?
3. How is the work related to what I want to do?
4. How does what I want to do build upon this work?

Note how (1) and (2) place the work in context of the grand scheme of things, and have nothing to do with a specific project. However, after answering (1) and (2), one is better prepared to answer (3) and (4), which now focus a student's particular work. The best place for crystallizing this information is typically the annotated bibliography, for each reference encountered. It may be useful in other places too.

It is also completely possible to answer (3) and (4) with "This work has nothing to do with my project" and "My project does not build on this work at all," respectively. This serves as a good way to explicitly scope one's project, and makes students realize that many readings simply build one's knowledge of the field, without necessarily having a direct influence on a specific project.[1]

Presentation

The goal of your presentation is not necessarily to cover the entire report, but rather to provide the highlights and most important points of the report. For this presentation, as in any presentation, start with your audience and identify what they are looking for. The real audience consists of the faculty members in computer science, especially those who are not your advisor.

Your presentation must quickly address the following questions:

- What is your topic area?
- What is the specific problem you addressed?
- What are the main related works? Use these to refine your problem even more precisely.
- What is your solution?
- Did you do something technically complex?
- Is it somewhat unique or new?
- Did you provide a contribution?

The challenges here are:

- Presenting the results of a semester-long project in 40 minutes.
- Presenting to an audience that is not familiar enough with your topic to readily understand the problem and your solution.

You need to find appropriate slices of the problem and solution to work through for the audience so that they can understand the slice and see your solution.

[1] Comments by Dr. Alfonso Cardenas at UCLA to Dr. J.D.N. Dionisio during the course of his PhD work at UCLA.

Unsuccessful presentations are caused by one or more of the following:

- A poor choice of topic. If you haven't found a problem or you've chosen a bad problem, you likely will give a bad presentation. In most cases, your advisor should not allow you to present without an acceptable topic.
- Trying to include your entire paper into the talk.
- Focusing on a tree and not making sure the audience is aware of the forest or alternatively, focusing on the forest and never getting to a tree. It is very difficult to present enough breadth for the audience to understand your problem and solution, yet also provide into enough detail to show technical merit. Thus, you must balance broad discussion with deeper technical discussions.

Great presentations often include some of the following:

- Show us an interesting technical problem that we can quickly understand, then work through the solution using your approach. The problem should exemplify your problem area, and the solution should exemplify your approach. If it is an interesting small problem, then this becomes a great presentation.
- Show us some interesting simple algorithm we may not have considered.

Finally, *be prepared for questions on your presentation.* Ask your advisor before the talk what kinds of questions the faculty are likely to ask.

Grading

The following is a general guideline regarding grades. It supplements, but does not replace, the information on grading provided in the CMSI/ELEC 601 *Overview* handout.

Grading Rubric

Final *project* grades are assigned as follows:

A	Student demonstrates a fully functional prototype
A-	Student demonstrates a partially functional prototype
B	Student has completed the research and design of the project but does not have a working prototype to demonstrate

Final *course* grades will be assigned as follows:

A-/A	Student has mastered the topic, contributed his/her own ideas, and produced a well-written paper.
B-/B/B+	Student has a good grasp of the material, but either does not present the material well or has contributed little in terms of original work.
C-/C/C+	Student has failed to grasp basic material or contributed little in terms of original work, but has made an acceptable final presentation and has produced a paper that thoroughly covers background material on the topic.

Plagiarism

Plagiarism, intentional or accidental, violates the LMU Honor Code and standards for academic honesty. It is not tolerated and is can lead to disciplinary action. Students unfamiliar with

writing research papers should attend the plagiarism workshop offered by the Learning Research Center and make use of the many resources related to plagiarism available on the Web. Refer to the Academic Policies section of the LMU Graduate Bulletin for additional information.

Why are you taking CMSI/ELEC 601?

The Master's Degree is an affidavit from the University that you are able to do basic research. Your performance in CMSI/ELEC 601 will determine whether or not this can be awarded.

Suggested timeline:

- Before the semester begins — Agree on proposal, references with advisor
- Every two weeks — Status report, updated timeline
- Week 1 — Turn in signed proposal and references
- Week 3 — Annotated bibliography due
- Week 5 — Introduction, outline due
- Week 6 — Process model due
- Week 12 — Draft due
- Week 15 — Final paper and draft of slides due
- Week 16 — Presentation