SI 650 / EECS 549 Project Advice

as told to students, by other students

Fall 2020

What advice would you give students on the course project?

Communication is key. Make sure to talk to your team-mate on all the ideas. Don't get too focused on making it look pretty. Iterate fast and if a method doesn't work ask for help. Please submit the project early not 2 minutes before it is due

Do annotations as early as possible

This course has a fair amount of work to do. However, that doesn't mean it is a hard course. Spending time on BM25, TF-IDF could help you survive from this course. Further, learning some fancy NLP algorithms and applying it to the final project is also very helpful! Overall, this is a great course if you want to learn about information retrieval or NLP fields.

If you are going to use neural networks, consult the instructor very early on.

"ANNOTATE ANNOTATE ANNOTATE! Do this early and often! Annotating your data will give you a lot of options for evaluating whether your system is performing well if you don't have any benchmarks. Plus, it's great real world experience for understanding how your relevance judgements propagate to your evaluation metrics. Ever had a weird google search that didn't turn out how you expected? There is likely some underlying metric driving that weirdness that was allowed to slip through.

Also start early so you're not stressed! I know there is a lot going on and you're burned out and need a break, but trust me the burnout won't go away at the end of the semester and it'll be twice as bad trying to push through it when you have a steep project deadline waiting for you."
The workload of the course project will always be larger than you expected. As you go deeper, you will find something you purpose to do but you are not able to, but also get inspiration from your classmates, professor and GSIs. It usually happens that the goals are set too general or huge. When my teammate and I wrote a proposal, we were so ambitious that we found more than 10 databases and wanted to combine them together. There were about millions rows of data. However, we noticed that it was too large for our course project. Even your goal becomes smaller, gathering data is a long process. The datasets you created may be very "dirty" that much pre-processing will be needed. Besides, the methods of evaluation are difficult to think of and far more difficult to realize. If you decide to do the annotation manually, be early!!! The task will be very demanding. Good luck!

Pay more attention to the information retrieval part. Try to find out if there is any existing library for other parts of your project such as image recognition and save time on this part.

Follow and work on assignments. They give you good ideas about doing your project.

Don’t give up. I had a feeling I am not learning until the end of semester. But I was trying my best. You will learn something based on how much time you put and your learning objectives. I am satisfied with my final project and it’s on my portfolio.

Find a good dataset for your project. It is not important it is a big dataset. More importantly, it should be accessible and interesting. Think about your previous projects. I used a dataset that I had scraped for si507 and this time I added a search engine on top of that project .

Did I mention assignments are important? They are your step by step guide in completing your final project."

First, start early and don't wait until the last minute. Second, I would say a good and well-completed midpoint report is very important, since in the final week, you have lots of projects to do and it could help lower your burden very much. As for the contents of the course project, I believe the topic is absolutely "free" for choice, and I encourage students to think of any topics that relate to the techniques learned in this course. For the methods, although it is important to have a baseline, but it doesn't mean the baseline should be beaten by any of the other methods you implement. I think the course project is more like a trial, or a practice on certain methods for a task, so enjoy the process, the result might not be that satisfying, but is still worth your effort.

"First advice would be to pick a subject that is interesting to you. In my case, I decided to do a project about video games because it is something that I enjoy. That really helped me with motivation. It also helped me with thinking from the user's perspective. In my case, I was doing a video game recommendation system using users' queries. Being able to know what a user of my tool (gamer) would want really helped plan and understand what needed to be done.

The second advice would be to think in advance whether you'll require expert knowledge/annotations and how you'll obtain them. I was lucky enough to have a couple of friends available (who are also gamers) that could help me with reviewing the answers in an objective manner. That may not be the case for everyone, so I would suggest coordinating that kind of work in advance.

The final advice I would make is to establish weekly/bi-weekly goals/milestones. It is very easy to postpone work and the everything snowballs in the end. You will likely still finish the project and do a good job without having a project plan, but you will suffer those last few days. I'd rather avoid those sleepless nights. "

Data source is very important for starting the project. You have to check your dataset carefully before implementing your algorithm. We have changed our topic for 3 times because the data source is not suitable for the project. For example, the scale of the data source is not enough; Data set doesn’t include the attributes we need, etc.

You should decide the direction of your project early. We planned to build a recommendation system before, but later we found that our system became a combination of search engine and recommendation system. Therefore, we have to add more features for search engine after project update.

Picking queries is also important for evaluating your system. For example, in our project, we used the Amazon products dataset in 2015, the products in the dataset are out dated. So when we searched for the latest product, we got nothing. We doubted the feasibility of our system when we saw this result, but after we checked the content of our data set, we found the reason."

Before you proposed an idea, please make sure that. You have the input and output data. The input data is sometimes easy, while the output data sometimes need more effort. You may need to use annotation and crowdsource.

Start early. And be sure to start with a baseline (preferably a method you have implemented in your assignment)"

Be early bird

Give a search on specific example with different models instead of just theoretical analysis

Give a guarantee by implementing simple baseline

Building a search engine for a special topic such as recipe and wine could be interesting. In terms of dataset choice, it is better that the dataset has more information like the users' ratings and profiles. So if you have time you could considering building a simple recommendation system.

Do a lot of research when choosing a topic. The best way is to start with a review article and gradually search for relevant literature and methods.

Data preprocessing is very important, such as noise removal, normalization, etc.
When implementing the algorithm, do not write code in a hurry. It is best to write a pseudo code first, and then write the code after the specific method is clear.

A single experimental result cannot prove anything, so we must do ablation analysis before drawing conclusions.

When writing the report or blog, you need to pay attention to both professionalism and readability to ensure that your ideas can be expressed accurately.

Try your best to avoid a lot of data collection and annotation. It takes a lot of time and isn't the most exciting. My project required a lot of collection so I spent a hefty amount of time doing it, and ended up wishing I had a little more time to focus on the actual meat of the project. For my project I ended up making a simple Flask app so people can actually interact with the system I made - I thought this was super rewarding and cool to see people using what I had made so I'm glad I made the extra effort to deploy the project. Try not to be overly ambitious, I know that's easier said than done but it gets really stressful and discouraging when you bite off more than you can chew. I would suggest making a more simplified proposal, and then once you finish all the tasks you initially proposed it's fine to continue adding to what you've done. I think sticking to a stricter timeline towards the beginning of the semester could've helped me a little bit - I wish I had taken the project update more seriously. Try your best to pick a topic you think is interesting and meaningful even if it's a little more challenging, it'll be worth it in the end!

My advice is to start the report early and always put enough comments into your codes. Homework and the kaggle competition are supposed to prepare you for the final project. Please please please find yourself a teammate or two. If you choose to do an individual project, you will have to do much more than you would expect. Don't make your life harder! Teamwork wins.

I personally did the final project solely, considering how the new online learning platform might fragment the communication among group members. However, I think this was not a good choice based on my experience. This was my first course in IR systems, and I believe I could have achieved and learned way much more throughout the project if I had some partners to work with. Final project allows you to utilize all the course materials you learned, thus having a group member can be beneficial for attempting various approaches.

The biggest advice I can offer is to start outlining your project as soon as possible. In doing so, break down the project into bite-sized tasks that can be done within an hour's time for each task. From there, start implementing your project as soon as possible. I know it can be a drag and it's human nature to procrastinate, but you'll really be doing yourself a favor by the end of the semester. It might turn out that your proposed model doesn't work as well as you initially intended and you'd like to try out some additional features/ideas. However, you won't be able to try these out if you didn't leave any time by the end of implementing your project.

Regarding finding the motivation to start early (because it can be hard, especially in an online environment): I think its easier to motivate yourself if you think about starting early as doing future you a favor.

what kind of project to propose: For this course, including ranking for your system is better than just doing classification problems (from my own perspective since you can learn and do from other courses). Try to work on some new topics if you can and make sure you have enough data or accurate data (at least you can clean it to an accurate status)

What pitfalls to avoid:
1) If you are creating ground truth by yourself, start early so you can avoid some bias. It would be better if you can have two annotations on one item and take the average on the annotation score.
2) For the baseline, try to use the worst method. You don't need to have a good number on that. Even though your evaluation metrics are very high on the baseline, you don't need to worry about it. Think about why it more important than numbers.

Make timelines: Thanks to David. He really gave us good guidance on that. It's better you can do data exploration, build a baseline for your model during or before the midterm. In this way, your finals will be much easier. How you can focus on trying more new techniques to improve your model.

Time: Spend less time on getting data but more time on different algorithms. It's better to understand why one is better than the other one. Through this project, you will have a deeper understanding of what we learned in class. And you might even learn some new techniques if you arrange your time better. Set aside some time to create an interactive interface. A simple one will also be enough.

When looking for the project ideas, you could find good examples in slides, do research about interested topic earlier to figure out its feasibility. Make the blueprint and stick to the timeline. If the dataset need manually annotation, complete it earlier cause it takes really long time and may need adjustment or supplement while adjusting the model. The annotation and adjusting the model spend more time on. Also, some of our peers chose to do a project related to image and video instead of text. So starting early will be good to learn and ask for help from professor and GSI. They are really helpful.

When we were thinking about group project, we had few ideas about what is IR and the specific methods, what we could do was to search online to see what kind of projects other people do. In the beginning, we had to determine what should be the final deliverable. Are we going to make a recommendation system? Are we going to make a search engine? So and so. Having a better understanding about available methods would save you tons of time when designing the project. It took us a while to fully understand the method and finally implemented it. Other than that, we also spent some time designing evaluation metrics. Don't worry about reports too much in the beginning. When we finalized the methods and finished the code. It took just one day to finish the report and blog. Making a solid timeline is also important, especially when each team member has separate tasks and you need to work on your part based on your teammates' output.

Working in a small group (two people) can be helpful. Having discussions in the progress can help fine tuning the methods. An individual project is very challenging.

If annotated ground truth is required, start the annotation earlier, it can take up more time than expected. Also, more annotated data is better.

Spent most time on building and optimizing the models. Finding an applicable dataset can also take up a lot of time.

Look up for related works after having a brief idea on the methods. Seeing what others do can help with coming up a more comprehensive plan.

Budget enough time for the report and blog post.

I would like to recommend them to take surveys and background researches on their projects, so that they can gain an overview on the data they are going to handle and learn from the projects that others conducted. Also, I would like to recommend them to go to the office hour to share their progress and concerns with their GSIs and Professor, and gain feedback from them. In addition, I will inform them that the time consumption on data gathering and manipulation will be much longer than them think. They should start early to finish this part.

In terms of making a timeline for things you want to get done, the single best piece of advice I have is to physically block out time in your schedule to work on it every week. This is especially important if you are working with a partner or group. Without set time every week to meet up and work or work on it yourself, it is easy to fall behind because the final due date for semester long projects always seem so far away. Another piece of advice I would give is to spend at least some time searching for various datasets to see how easy it will be to find and/or preprocess data that will be usable before starting your project just so you know early on if your project is feasible or if you will need to switch. A last piece of advice just in general for the project is to try and pick an idea or project that you are genuinely interested in. If you are actually interested in it, you will be much more inclined to work and make progress on it regardless of the difficulty. Choose an item of interest, which will help you complete it well. Before deciding on a project, complete as much as possible to understand the various methods and uses of information retrieval (browse the courseware and the examples on kaggle). It is important to have a complete feasibility assessment for the project. Start early to avoid project failure due to too short time.

I will suggest that during doing group project, choosing an interesting topic and good teammates is important. There are lots of different aspects of information retrieval, and you can choose your interested topic.

Personally, coming up with a topic for this and other classes has been part of the challenge, and I tend to spend too much time in this stage. So, brainstorm topic ideas asap.

Discuss your ideas with the teaching team. And try to understand the technical implications of your project, and the feasibility of what you want to do asap.

Sounds counterintuitive but you need to have an overall understanding of the class, yes before you even start. This will help you visualize the stages needed to complete the project. Again, try to talk to the teaching team, or students who already took the course. Also, try to look ahead in the topics to understand the goal of the course, and what major topics will help you complete your project.

Collecting and processing the data is usually if not always the most important and time consuming part. Start early, the first week if possible. I think most people end up scraping data from websites.

This project may require that you use different coding skills to connect the pieces of your project. Review those skills if you are not sharp, maybe flask, django, html, scraping, pandas, etc.

If you want to work with a team, find one early, the first week if possible, yes even before the project is even discussed in class. And make sure that every member understands the goal, the problem, the technical requirements, the challenges, and every single stage in the project. It is OK to break down the tasks by strengths, but make sure that everyone has the idea of what the others are doing.

Make sure you have enough time to work on the assignments, because the assignments will be conducted in-class competitions;

Though the project proposals are due in a short period of time, there is plenty of time to work on the project. I think the key to a successful project is to incorporate the knowledge acquired in the lectures into the project without sticking to the content of the proposal.

Due to the wide range of content in the lectures, more time may be needed to fully master the content of the lectures."

Please start early if you deal with image data! At least run once to estimate the running time. Try to run your code both on Colab and Jupyter Notebook. Jupyter Notebook might be faster than Colab! And be careful of the disconnection problem when you run your code on Colab overnight! Good luck and have fun!
Short answer: under-promise, over-deliver. One of the main issues we ran into was trying to live up to our proposal. I think it's better to propose something slightly looser rather than something concrete that might not be attainable. Small increments in progress will get you to the finish line.

In terms of time allocation, I would definitely be mindful of the data you collect (if you're scraping data, really). We ran into some class imbalance when sampling our data, and didn't really take into account the distribution of these. I would also spend more time in high quality manual data annotation, if your project requires it. We spend some time in the last stretch manually annotating data to train a entity extraction model using spacy.

There were also so many routes to take, that it can be overwhelming to pick and choose what to do. Also, reading advanced papers can seem overwhelming, so don't feel the pressure of creating something groundbreaking; small steps are the way to go, build something basic, and iterate/improve.

Also, always take it as a learning experience. Overreaching can be useful, but always have a ""doomsday scenario"", if all else fails.

Start early

Make sure there is plenty of available data for you to do your proposed project

Attend Office Hours

From my view, the assignments were excellent and could help us had the deeper understanding about the course materials. So, my suggestion is that you could let students try more things about the information retrival via assignments or practice problems in class.

Definitely make sure you do thorough EDA on the data you’re working with before you start doing any type of system creation or machine learning. Planning things based on unfounded assumptions on the data can give you a headache when you find out that those assumptions weren’t true.

Get the most you can out of office hours, ask to confirm your approach or for alternatives or anything else. It is probably the going to save you a lot of time and improve the quality of your work.

My group's biggest regret is that we reached the end of our implementation only to then realized that, when it was time to evaluate performance, the example queries we did relevancy annotations for (establishing ""ground truth"") were all too narrow/specific and mostly aligned with document titles rather than encapsulating terms that would appear in the body text, which was a focus of our project. This meant, to do any real meaningful analysis of performance, we would've had to have redone annotations in record time.

I don't feel I've ever had an undergraduate or graduate course that included that kind of time-intensive annotation element and we underthought on it out of inexperience, and that easily had the heaviest negative impact we could've easily avoided.

TLDR: Think about your example queries very carefully before annotation, because it's a major time investment that directly influences your ability to measure performance and is basically the last thing you want to have to revisit / redo and probably is also the easiest thing to under-think because you probably haven't had to do anything like it before.

It's important to collect enough labeled data or know where to collected labeled data before doing a search engine project.

Take notes for all of the information you have looked up for solving problems for the assignments. A lot of time, those questions you searched for became great resources when you are trying to come up with your own project.

Keep an open mind to search for a good/cool problem statement. You need not build another google, but a search engine which helps a community or an institution to get what they search for is good enough. While doing projects keep in mind the expectation of the course and do not hesitate to explore new things. Be thorough with the related work, understand how others approach and how did they evaluate what are their limitations and future scope. These are very minute details but help a lot while doing projects. Sometime we think that deep learning would give us better results but it's not the case always. We might fall short of data for deep learning. Spend good amount of time doing the project along with course work. There lies a great learning opportunity along side the classroom learning. David is a real good Professor we have, he encourages you to explore, which yields good results. Attend office hours when in doubt, ask others in breakout session when you're stuck. Talk to people in different forums for advice.

Start the project early (literally best advice ever and i know you're reading this and won't listen, because i did the exact same thing). Even though I know you won't take my advice, but it is important to apply what you've learned in class during the semester to the project. At first i thought ""oh i'm only learning the basic stuff, i want to do complicated stuff for my project."" this is the wrong mindset. my project ended up using NDCG to evaluate my results, and it is way better and easier to implement than something like f1 recall. don't think that you can only work on the project after finish all the course materials. you can literally start your project after second week of class.

Spend time on assignments and mid term. This is how you learn (again i know probably 100 other people repeated this 100 times), but this is the way to go, that's why everyone is saying it.

work on something fun and you're passionate about for your project. I worked on Formula 1 (relatively niche sport in North America, but big everywhere else), and i absolutely enjoyed it, even if it meant reading 100+ pages of rules and regulations. Your passion is what drives you and motivate you to finish the code and you will be proud of what you accomplished in the end. The code is amazing :)

Do readings before class and spend some time to dedicate to this class. most effective way to learn. If you don't know stuff, google. "

First of all, I personally think that it is important to conduct some background research before making a proposal so that you can have more ideas. Moreover, you can know how difficult your idea is. Then, it is very helpful to have a timeline schedule, and it would be better if you can set up a monthly plan and a weekly plan. In addition, I recommend that you set aside at least 1 hour a week for meetings with your teammates to update your ideas and completed work. Besides, tuning hyperparameters and training models usually takes a fat lot of time, especially if your dataset is very large or you need to use some matrix-based models, so you can have a sample dataset to do these steps first. Finally, when you are looking for datasets, choosing those with ground-truth will greatly help you save a lot of time and energy, and make your experimental results more convincing.

Knowing some basic Linux commands (such as stdin, stdout, cat, grep, cut, tr, ...) can be helpful when working with large datasets.

Having some understanding of Python classes and objects would be helpful when utilizing packages such as metapy or gensim (if you are familiar with classes in other programming languages, you are probably fine).

When building a search engine, don't be afraid to write your own implementation. Using Python packages does not make things a lot easier -- especially metapy, which does not work well with a Windows machine or the Flask framework, and does not have very detailed documentations. "
If you're doing a vertical search project: If I were you, I would quickly get through preprocessing and ranker setup so that you can spend more time on evaluation and iteration. I found that after my initial evaluation was where I found interesting things to work on! Best of luck!

One advice I want to provide to our next round of student in doing project is think big and plan ahead. Information Retrieval has a very broad of applications. Thinking big means finding out what interests you the most so that you would be motivated to do it. Also, plan ahead is very important. My team and I was passionate about building a search engine for all data science project/datasets. However, when the proposal deadline approached, we still had no idea how to start such a large ambitious project with limited resources and knowledge. Luckily, we reached out to the instruction team and properly rescoped our project. And we were very happy about the final project's turn out.