Ling 506: Affective Computing

Term Project Proposal

Due date: Oct. 15th, 2020 (by midnight)

NOTE: THIS IS NOT MY ACTUAL PROPOSAL

You are required to submit a one- or two-page written project proposal. The proposal should include enough detail to make sure you've found a good problem, you understand how hard it is, you've thought a bit about a general plan for how to attack it, and you have an idea about what kind of experiments you might run to test the success of your implementation.  Please do not be vague in your written descriptions. You are given the following outline (questions to answer/address) to help you in the process:

* Goal (Problem definition and importance)
  + What am I going to do?
    - My hypothesis for this project builds on an assumption of how the human brain work at a more abstract level. In sociology, people have social roles (ie, student, friend, employee, etc.). These social roles effect how people behave in different contexts. It affects everything from their communication style to how they express emotion and the intensity of that emotion. Through predicting the domain of a text of a text using topic modelling, it may therefore be possible to produce a compromise between leveraging the benefits of specialized corpora, while also having a generalizable application. Questions remain to be answered also about the overlap between domains, few papers have ever tackled this head on. It is likely that the way emotions are expressed will be similar enough between some domains that they can be grouped together into a broader domain in the topic modelling, however, as has been shown in previous works, there must be a bounds on this, as an all-inclusive generalized corpus will perform poorly. Another possible extension of this social roles hypothesis is that people will draw from the knowledge of language that they have developed in the generalized corpus that brain builds over your life, and then relate to the specialized role for specifically how to react. Thus, it may be beneficial to also test large, generalized corpora as combined with specialized corpora in a weighted approach, where more focus is given to the predicted domain, but the generalized corpus is still included to increase test size. My goal is to implement a topic modelling algorithm with various combinations of corpora to analyze overlap between domains and the potential to leverage specialized corpora in a general task. As not all combinations will perform well, in fact many will not, my goal is to be thorough in testing combinations such that regardless of outcome, important conclusions can be drawn from the results.
  + Who would benefit?
    - The people who stand to benefit the most from this research are those who wish to have a generalized application without the ability to leverage the compute intensive, large, generalized models. While not a sentiment model, in doing research recently, I’ve found that to even work with GPT-2, it can require graphics cards costing over a thousand dollars, and to fine tune the algorithm, it requires a workstation an order of magnitude more expensive. GPT-3 is much more compute expensive, as is the case for applications such as AI Dungeon, which struggle to afford its use. Even if the results of this research do not outperform state of the art models, if the process results in performance gains over specialized and generalized corpora, it will be a successful proof of concept, as this process of topic model prediction can be integrated into current and future state of the art models that are prohibitively compute expensive for me to test in this project. Also standing to gain are those of people who work in specialized domains, where there exist very few specialized corpora, or corpora of small sizes. If it is shown that a weighted approach to combining the generalized corpora with specialized domain corpora is beneficial, then researchers will be able to integrate this process with any corpora or computational model that they choose.

Problem Difficulty

* + Why is it hard?
    - This research is difficult because the connection between corpus size, corpus specialization and performance are not clear. While previous studies have combined corpora within the context of their research, and other studies have tried to tackle the matter in testing, nearly all works that I have found as of 2020 do not publicize enough data to draw conclusions from their trials or corpus, and for those that have, their results have come up short of a definitive answer/solution. Every paper that has attempted a similar problem has failed or come up short. However, each time, a new insight is drawn, and the community gets a little bit closer to a working model.
  + How hard is it?
    - The project has a flexible difficulty, ranging from very achievable to impossibly difficult, depending on how thorough I am in my implementation and testing. An important part of this project will be to be realistic with myself about what I can personally achieve given my own time, abilities, and compute resources, and scaling my project accordingly. However, I believe that I can produce a useful and meaningful paper from this regardless.
* Previous Work [this bullet point is NOT mandatory now, but recommended]
  + What have others tried?
    - An Analysis of Annotated Corpora for Emotion Classification in Text
      * combined a number of corpora together with different annotation styles in an effort to produce a better, unified corpus. The paper failed to produce any gains in results, but proposed transfer learning and domain adaptation as potential future work, which is similar to what I am doing.
    - GPT and other transformers
      * tried (and succeeded) to create a massive, generalized corpus and fine tune it with specialized corpus. While not a sentiment analysis model, and certainly too large for myself to even work with, it demonstrates that there is a method through which a generalized corpus can be leveraged to make a specialized corpus perform better.
    - Assessing the Corpus Size vs. Similarity Trade-off for Word Embeddings in Clinical NLP
      * Explored an intermediate compromise between choosing small, clinical corpora and large, less representative corpora. Showed benefit in combining certain specialized corpora for clinical NLP, mentioned the need for future papers that evaluate on more corpora to measure intra-domain variance
    - In Pursuit of an Efficient Multi-Domain Text Classification Algorithm
      * An excellent paper in analyzing how to achieve a generalized approach to text classification by combining corpora. Provided excellent results and discussions in how inter and intra domain variance affects performance as different domains are combined. Implemented simpler models such as decision trees, naïve bayes, SVM, and maximum entropy markovization, which are models I would be able to run. Failed to solve either problem that it sought to solve, but provided key insights into making text classification work in multiple domains. Recommended exploring paragraph structure and greater semantic knowledge.
    - Frustratingly Easy Domain Adaptation
      * An old (2007), but useful method for doing domain adaptation and multi-domain adaptation which as seen in the paper presented in class, can result in better performance than simply concatenating corpora together.
    - Efficient Topic Level Opinion Mining and Sentiment Analysis Algorithm using Latent Dirichlet Allocation Model
      * An recent model produced last year that uses a similar process to what I propose in this paper. It uses LDA topic modelling to predict a topic, then it classifies the topic using SVM or one of several neural models.
    - Leveraging Multi-Domain Prior Knowledge in Topic Models
      * Proposed their own model MDK-LDA, to use prior knowledge of multiple past domains for producing better topics in the new domain.
    - LDA-based Topic Modelling in Text Sentiment Classification: An Empirical Analysis
      * In a fairly thorough methodology, they conduct a literature review over the use of LDA topic modelling in text sentiment analysis and run tests on how models perform, as well as constructing various ensembles.
    - Enriching Topic Coherence on Reviews for Cross-Domain Recommendation
      * Use topic modelling methods to extract user content like reviews, and combine the method with semantic coherence techniques to link topics between domains, using Wikipedia as the original source.
    - Challenges and Recommended Solutions in Multi-Source and Multi-Domain Sentiment Analysis
      * Explains and recommends solutions for the problem I am working on.
* Approach
  + What approach am I going to try?
    - In doing previous work research as I was writing this proposal, I discovered that there is more research on this topic than was previously thought during my original project meeting. After discovering more research on the LDA approach to sentiment analysis, finding methods for domain adaptation, and the plethora of other studies and methods, it seems that I will already need to reorient my project. This approach may change again later on, but as of right now, I think that the best route to go with this project is to focus more on what separates it from previous studies, and to stand out with thorough data representation and explanation of results. It seems that the LDA approaches and others proposed recently have been very mathematical and probability based. There seems to have also been a misunderstanding in my initial research, as in topic modelling, topics are expected to change through the document, and these topics exist within a domain. However, I think what I want to focus more on is rather, a domain level/social context prediction in that I can build on top of these other features. Imagine one is speaking to a friend, topics will change throughout the conversation, and domains will change as you and your friend go to different areas (ie, at a park vs at a restaurant). Current methods may be able to roughly predict the current topic of conversation based on unsupervised mathematical probabilities, and there may be some scarce research around predicting the specific domain that you are in, but there is nothing around the social context level prediction. As far as I can tell in the numerous papers I’ve read at this point, this is actually unique.
    - Due to the uniqueness, it is extremely unlikely that I will find any “social context” labelled data sets, thus, my approach will be to identify sources of corpus data from sources with relatively consistent formality/friendliness/etc. where it would intuitively allow good data. Popular professional accounts and politicians on twitter may be a good source for one category for example. Another may be to include journalism (but also accounting for variations in publisher styles, ie, New York Times vs Huffington Post). I could also use subsets of the Reddit Corpus which were used to train the bots on r/SubredditSimulatorGPT2, as if there is enough data in certain communities to train GPT2, then there is likely enough for my purposes. Note, many of these corpora will NOT be annotated by sentiment, polarity, or anything. Instead, this will be part of something more like an aspect classifier, where the aspect is actually the linguistic elements of formality/friendliness/etc. I will use this as a preprocessor for a model, which will separate all the data before it is trained into models
    - A possible resource for this may actually be sociolinguistics papers, as a source of citation for possible domains that have been previously ignored.
  + Why do I think it will work well?
    - It has been shown that specialized corpora are better for performance, and large corpora are better, however, most times you must choose one or the other. It has been shown previously that if one chooses the proper corpora, it can improve performance to combine corpora.
* Methodology
  + What resources (e.g.: data, annotations, etc.) are required and how am I going to get them (unless I already have them)?
    - For this project, I will need to gather various corpora from different domains. To test the importance of subject relation, I should have a massive generalized corpus of all English, a broad corpora of something such as formal English, and narrow down the corpora repeatedly. Due to the nature of my project, I will gather as many corpora as is practical to work with.
  + What computational models do I have in mind to solve the problem?
    - Because the nature of my project is related to topic modelling and corpora, my project serves to enhance any existing model. I would like to test various models, including ones that are as state of the art as I am able to handle in terms of hardware, and in terms of being able to implement. Particularly, I would like to compare between models that perform well with small corpora, and models that perform well, but only with a large corpora. Essentially, I will selectively choose models with different training size requirements, different scaling of accuracies as training size increases, and ones that are affected differently by variance within a corpus. I believe that my technique may allow for large corpus dependent models to leverage the advantages of having a specialized corpus. I also expect to see decreases in performance when my process is applied to models that are more strongly influenced by generalized corpora. However, these models should still be tested, as if I am able to minimize their performance loss as corpora is generalized, it may be a good intermediate compromise in some use-cases.
  + Which aspects of your model(s) are particularly hard?
    - The difficult part in implementing models will be finding those which have enough detail/published code to integrate into my research, as well as finding a way to run as many trials as possible. Due to limited compute resources, I may struggle to run as many tests as I would like on the variations of corpora and models. As I learned in Ling 406, features that show improvement in one use-case may not be beneficial in others, and so I need to make sure to test on different types of models to determine which models, if any, that my process improves on. In a previous study from 2016 that I found, they mentioned that a limitation of their paper was that they could only run their tests for a week of compute time, but there is certainly no way that I would be able to run a program for a full week straight, and it is also likely that they were working with better compute resources than I. One possible solution for this would be to cross-reference existing graphs of how models are expected to scale according to training size, variance, etc. Then I could produce a graph that maps out my results, and then traces a prediction line as to how those results may improve if more trials were able to have been run.
  + What to do if the hard steps don't work out (meaning, what is my plan B)?
    - If my hypothesis is wrong/doesn’t work, my plan is to thoroughly document and offer a discussion on possible reasons why, with encouragement for future research. As found in the unified corpus paper we read in the beginning of the semester, a research paper can still prove useful to the public if it has provided enough detail and insight as to what worked, what didn’t, and what could be improved.
    - If the process of topic modelling proves to be too difficult, my plan is to hard code the program so that it chooses what I believe are related domains, after all, the research contribution here lies in the process and proof of concept, rather than the underlying implementations. It is unlikely that I would be able to outperform SOTA topic modelling within our time frame regardless. It will then combine the corpora in unique ways, and I will draw conclusions on the results and intra-domain variance.
    - If one or more aspects of my proposal don’t work out for one reason or another, my plan is to shift my focus over towards providing as comprehensive of data as possible on how different corpora compare, analyze intra-and inter-domain variance across corpora and combined corpora, analyze the effects of combining corpora, provide discussion of results and insights, and describe potential future work, as the published papers before this one have done. If it came down to this, I may orient my paper also as more of a literature survey on this topic.
* Metrics
  + How will I measure success?
    - I will measure success by creating color coded tables comparing the different combinations of corpora and different implementations to each other. I will be running my program through a series of trials with various combinations of corpora. I will be looking for trends as to what domain combinations work well together, and I will also be comparing how the model performs with different weights of combinations of a large, generalized corpus with specialized corpora. Success will be measured in accuracy, F1, precision, and recall, with tables published for each model in the paper and a spreadsheet of trial runs included in the public Github that I will produce for this research. However, for the purposes of a cleaner paper, I will likely include only the comparisons of accuracy or F1 in the paper, unless there is some significant difference found between other metrics that is worth noting in the paper (I will decide on F1 or accuracy depending on which one I find is more used in past publications of the models I implement, as I intend to cross reference performance of past papers in my discussion)
    - It will also be important to note inter- and intra-variance between corpora, both as being useful for researchers to decide which domains and corpora they should combine in their use case, and corpus variance may even be useful in the process of deciding which corpora to combine for a topic.
    - Another metric that I am considering to include as a bonus is the runtimes and data memory related to running these models (along with the source code, such that a more experienced algorithms expert would be able to distinguish between possible differences in my personal implementation’s efficiency, and the efficiency of the models). I will record these as a way for myself to predict how my project is scaling as I run my various trials, but I might not include them if I later feel that it is simply distracting from the paper. If there is a noticeable gain or loss in performance, or anything unexpected, I will research and discuss it further, as this either denotes a finding in my research, or an issue with how I am running my program.
* Summary
  + Why is this project of interest to me and what will I learn by doing this project?
    - I’m interested in this because I am personally a big fan of digital assistants and AI companions, as well as working with NLP analysis in social contexts. Time and time again, we’ve borrowed concepts from our knowledge of the brain, as it is even the foundation of artificial intelligence, and so I think that we could use more techniques that leverage the mechanisms of the brain and mind, not just on the neurological level, but also more abstracted such as in sociolinguistics.