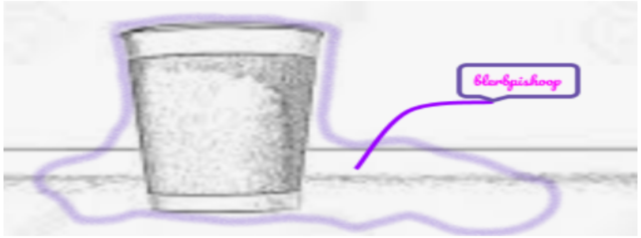
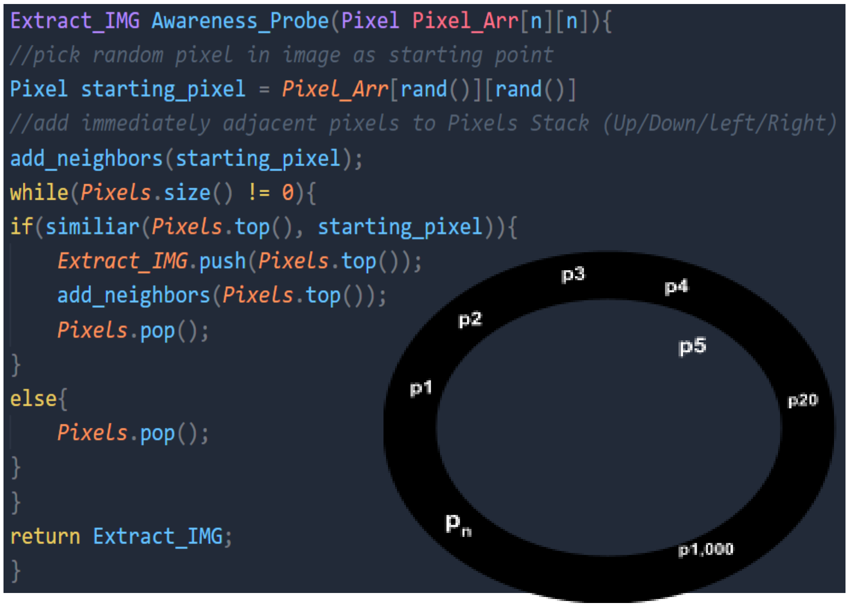
Personal Statement MA Linguistics

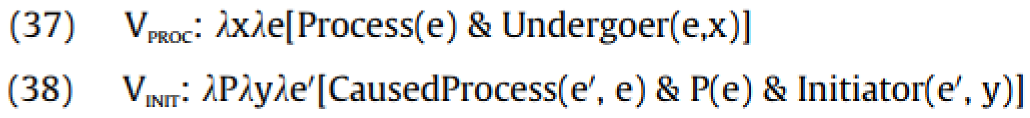
I’m interested in working on what I call the frame problem, which lies at the intersection of philosophy and language learnability. The frame problem specifically refers to how any human decides to extract and categorize certain subsets of a visual scene in the way that they do. Language is essentially a set of categories which are obtained by children, at some stage in the developmental process, by making very precise and universal visual abstractions. Languages can vary drastically in how they embed these categories in syntax, but the underlying semantic categories being expressed are by and large universal: every language has a word for leg, cat, and chair etc (ignoring obvious variations in cultural contingencies). Given any continuous array of $\boldsymbol{N}$ pixels enclosed in a finite region, there exists 2^N possible visual discriminations one could make where each pixel is either included or excluded. The vast majority of the visual bits pulled out of this scene would not be pragmatically or conceptually meaningful. Take a simple example of a cup sitting on a table. While this observation may seem absurd, it is not necessarily trivial that there is no existing human language that has a word to describe the cup plus some amount of area of the table immediately around the cup. The human mind is able to seamlessly treat the cup and the table as separate entities despite them being part of the same scene. There must be underlying cognitive features, likely deeply built in the human visual system, that prevent such categories from being entertained in the first place. Immense progress has been made in the field of visual processing but it has largely occurred alongside improvements in statistical tools and computers ability to analyze large swaths of data. My goal is to design an algorithm that is able to create visual discriminations of the sort above on the basis of very minimal amounts of data: it won’t be comparing the image of the cup on the table to thousands of previous images of cups on tables in slightly different positions. A rough outline of the sort of direction I would like to take the algorithm in is detailed below.





```
Extract_IMG Awareness_Probe(Pixel Pixel_Arr[n][n]){
//pick random pixel in image as starting point
Pixel starting_pixel = Pixel_Arr[rand()][rand()]
//add immediately adjacent pixels to Pixels Stack (Up/Down/left/Right)
add_neighbors(starting_pixel);
while(Pixels.size() != 0){
if(similiar(Pixels.top(), starting_pixel)){
    Extract_IMG.push(Pixels.top());
    add_neighbors(Pixels.top());
    Pixels.pop();
}
else{
    Pixels.pop();
}
}
return Extract_IMG;
}
```

Starting at pixel p1, we would expand the awareness probe by successively adding neighbors of p1 and then neighbors of those neighbors. The way we would extract a black circle from a white background, for instance, would be to only add pixels similar to *p1*, thus rejecting *p5* and adding *p1* through $p_{\mathbf{n}}$. In this scenario, *similarity* is extremely simple in that the image only has a distributional variation of 2 = {black, white}. The awareness probe expands by adding pixels to the sphere of awareness, which functions as a screenshot in the end, extracting any possible shape or bits of visual elements from the scene, conserving the proportions and angles of the extracted image. The code above is a very rough outline of the central idea behind how I hope to tackle image extraction in the context of language learning. It is important to note that while I will be using the word pixel, I merely mean a small area of space roughly like an ***ϵ-ball*** containing any number of possible visual features. The actual version of this model would add neighbors continuously instead of discretely, where the size of the ϵ-ball's would be determined by the relative granularity of the image being analyzed. The core of the work would be in developing a very expansive and general purpose functor ***Similar(...).*** *Similar()* needs to be expanded by being able to compare pixels on the basis of many additional visual variables such as color, depth, shape, lighting, texture, and a multitude of other possible discriminatory visual characteristics. Once *Similar()* has a vast set of features to use, we can combinatorially iterate over the feature set and continuously redefine what we mean when we say two pixels are similar. For a feature set containing {orange, black, red, hard, soft, rough, smooth}, *Similar()* could produce a vast set of extractions from the image by merely varying which subsets of features it chooses to use in defining similarity. For example *Similar(...[Black⊕Smooth]...)* would extract a very different image from the scene than *Similar(...[Black⊕Smooth⊕Hard⊕Red]...)* would. This algorithm attempts to capture how humans normally interact with visual scenes, by scanning the scene and focusing on bits that share some similarity, and extracting them as unified wholes from the scene while ignoring elements deemed different. Additionally, I want to work on using lambda calculus to increasingly broaden the number of semantic types within single syntactic categories. This is essentially what is done in the work of Gillian Ramchand and Peter Svenonius **[2]** on the core functional hierarchy (CFH) where they begin dividing the space of syntactic categories into events, situations, and propositions. In the short term I want to add to their work by creating semantic denotations for the class of prepositions and adjectives in lambda calculus along the same lines that they did for verbs. Examples of their unique semantic denotations of various verbs in their verb hierarchy ($v_{EVT} > v_{INIT} > v_{PASS} > v_{PROC} > v_{RES}$) is below:



(37) V_{PROC}: $\lambda x \lambda e[\text{Process}(e)\ \&\ \text{Undergoer}(e,x)]$

(38) V_{INIT}: $\lambda P \lambda y \lambda e'[\text{CausedProcess}(e', e)\ \&\ P(e)\ \&\ \text{Initiator}(e', y)]$

I took example sentences from *A Hierarchy with, of, and for Preposition Supersenses* **[1]** to give a flavor of what this would look like. They identify differences between prepositions that merely accompany the main event and prepositions that actively participate in the event structure.

(1) (Accompaniment) Tim prefers [tea **with** crumpets]

(2) (Joint Participation) Tim sat **with** his friend (*tim sat* can standalone)

$\mathbf{P}_{\mathbf{ACCOMP}}$: $\lambda n_1 \lambda n_2[n_1 \cap n_2 = n_3]$ here the preposition functions to create a joint noun but does not modify the event structure in the way P_{JOINT_PART} does.

$\mathbf{P}_{\mathbf{JOINT\_PART}}$: $\lambda Q \lambda e[Q(\mathbf{v}\langle\mathbf{x}\rangle) -> Q(\mathbf{v}\langle\mathbf{x,e}\rangle)]$ where e = *Tim* and Q is an event containing an intransitive verb made transitive by the preposition which accepts an entity(*friend*), animate or inanimate. Examples such as the one above illustrate that the syntactic category of preposition might indeed be too broad a category and that we are mistakenly abstracting over semantic differences speakers store in their knowledge base. The general goal of natural language processing should be to increasingly broaden machines' semantic understanding which requires, in the short term, to make moves that may appear on their face unparsimonious. Expounding on individual categories such as adjectives and prepositions may seem cumbersome and lacking in generality. On the other hand, it is likely that the cognitive underpinnings of language house such expansive and intricate semantic distinctions that rely on perceptions of argument, event, situation, and propositional structure. Creating distinctions such as $\mathbf{P}_{\mathbf{ACCOMP}}$ and $\mathbf{P}_{\mathbf{JOINT\_PART}}$ is useful in the sense that it adds salient semantic information to items we previously would be treating in the same manner in a syntactic framework. This additional semantic information can then be used to bootstrap into higher level semantic distinctions on the phrasal level for structures containing these elements. A computer being asked to model an image of the scene described in sentence (2) would need to leverage this semantic distinction between $\mathbf{P}_{\mathbf{ACCOMP}}$ and $\mathbf{P}_{\mathbf{JOINT\_PART}}$ to correctly depict the friend also sitting down. If it were to treat the preposition *with* as in (1) it would just depict: Tim sitting, with his friend also in the scene, maybe standing beside him but not necessarily also sitting.

Citations:

[1] Schneider, Nathan, et al. “A Hierarchy with, of, and for Preposition Supersenses.” *Proceedings of The 9th Linguistic Annotation Workshop*, 2015, https://doi.org/10.3115/v1/w15-1612.

[2] Ramchand, Gillian, and Peter Svenonius. “Deriving the Functional Hierarchy.” *Language Sciences*, vol. 46, 2014, pp. 152–174., https://doi.org/10.1016/j.langsci.2014.06.013.