# Topics

### Natural Language

- Automatic Natural Language -> Temporal Logic (there is a paper documenting this, worth exploring)

- Natural Language Specifications (there are papers in this area too, worth exploring)

### Language Learning

- something paralleling the idea that funblock is doing? (i.e: rewrite rules to teach coding, maybe something from models to teach foreign languages?)

- kripke structures and language modeling?

- CTL and language modeling? Tie in with the concept of linking together conversation flows in order to create something to help language learners? (Here I guess we differ from language modeling perhaps in that we care less about the correct syntax and order of words, but rather generalize to a concept of a “flow of conversation”?)

### Floating Thoughts

- There is a PPT on automatic detection of read-back errors having to deal with airplane control and instructions for landing. There’s a section here that says, “key decision for automation: is the read back error worth alerting?” And I think that this type of “read-back” concept could probably be applied to language learning, where the automation decides whether or not it is worth correcting your speech? There’s probably something noteworthy here about how we can use the model checking methods (automated) that we’ve been looking at throughout the semester to detect abnormalities (errors) in constructed speech. Probably worth formulating a proposition

# Formulating a Block

### Considerations

Interests: Language Modeling, Language Learning, Language Theory

Links: CTL, automated model checking, kripke structures

Tools: automated read-back error checking? conversation flow checking/modeling or generation?

### Read-Back Proposition

Links: Automated model checking, CTL, kripke structures.

Tools: Constructed speech error-detection in natural language.

Proposition: “Readback error” is a concept in aviation safety when the pilot has to contact the tower with regards to information on flight patterns concerned with landing. Taking inspiration from the notion of model checking as well as read-back error detection, we apply this concept to language modeling/constructed speech to detect errors (grammatical or colloquial) made by users in a language learning environment. This could be applied to a tool to assist a User to get feedback on their constructed speech without necessarily having another person present. We can treat the concept of a conversation or a constructed speech as a system/model that we can then check (or create automated checks) via the methods we discussed throughout the semester.