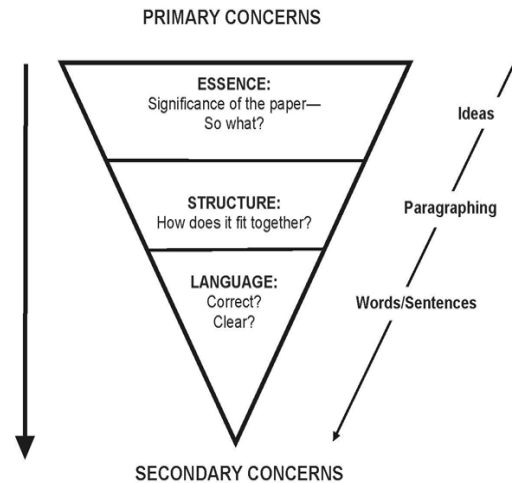


Proposal Group: Balkind

Readers' Name: Katy Tsao

1. Overview: Read through the entire proposal. Does this paper fulfill the assignment? Can you see what the writer's project is? Does it speak to its intended audience? Give your initial reactions here.

Yes, I think it does satisfy the assignment and I understood it pretty well



2. Now, go back to the beginning of the proposal and identify the main research problem and main contribution of the proposed work, and write them below. This should be in your own words.

Main research problem:

Git diffs, as they are now, show any difference in a line of code, even if it is just stylistic changes, which makes diffs difficult to read

Main contribution of proposed work:

They will work with GumTree, which uses syntax trees to determine what code is actually being changed, and test it to make sure it gives the expected output

3. Next read the introduction and the approach section if needed (again) carefully for the motivation. Does the paper clearly motivate the problem and make you see that this is an important problem? Do you understand how the proposed work will build on previous solutions? Give feedback below.

I know it is important because as a CS major I have had to read through long and confusing diffs, and a program that only shows differences in actual code would be super helpful. It seems that they are building on a previous solution by testing a proposed algorithm and making sure it works as intended

4. Next, read the proposed approach section and look for the technical details of the proposed work. Do you understand what the group will do? Do you understand how this work will solve the problem? Are technical details clearly motivated with the problems they will solve?

They never actually defined AST which I think would be helpful in understanding the rest of the paper. Other than that, I didn't really find many technical terms that I wasn't familiar with

5. Next, read the proposed evaluation plan and timeline. Does it seem reasonable and appropriate? Do you have any concerns?

It seems appropriate, although I do not know how long it would take to become familiar with and test GumTree

6. In terms of style and mechanics, is the writing easy to read and follow logically? Are terms defined adequately? Are citations used correctly? Are there spelling, punctuation or usage errors? Is there a pattern to these errors? Feel free to mark some changes in the document.

There are a few missing words but it is easy to follow logically. They could maybe even consider removing some of their repeated ideas like they kept bringing up the fact that a git diff would show spacing as different

7. What are at least two things you think are particularly strong in this draft?

The problem and solution were spelled out very clearly  
There was a good balance of jargon and normal language

8. What are at least two **specific** suggestions for changes you'd like to see in the next draft?

Define AST before using the acronym!!!  
Add images to help your reader understand with visuals