**Title:**

Hello, I’m Bryan, and my project is on the AP Art portion of the course: drawing geometry diagrams from their problem statements.

**Motivation I:**

Here is a typical olympiad-level geometry problem. I’ve taken the liberty to bold all the typical elements -- a central triangle, some sort of circle (in this case, the circumscribing circle of ABC, or the circumcircle, among others), and a lot of special lines and points, such as the angle bisectors and the midpoints.

The first thing we need to do to actually try to solve this is to draw the diagram for this problem - after all, it’s very difficult to solve a geometry problem without drawing a picture for it. As this is the AP Art portion of the course, I’ve drawn one up.

**Motivation II:**

Here is a quick hand-drawn diagram for this problem. The composition is okay - the circles for the most part look like circles, and the lines are generally straight. However, if you’re actually trying to solve the problem, it would help if we could just look at the diagram and discern a statement or two that looks true, just based on the diagram. This hand-drawn diagram isn’t good enough for that purpose. **(play animation for diagram)**

In contrast, here is a diagram generated by a computer with the Asymptote language. This is much better and cleaner, and there are some facts that are true in general that might be easier to see, given that everything is drawn precisely. For example, it may not have been obvious before, but lines AD and NM are actually parallel, and we can actually prove this.

**Motivation III:**

We might also see something going on with the second intersection of circles ADM and ABC -- in particular, it might be that this unmarked point is actually on line ML! Spoiler alert: that’s actually true in general, and it’s part of one possible solution to the problem.

As an added benefit to having these computer-generated diagrams is that they are also much more versatile. For example, if we want to add this second intersection into the diagram, and call it X, we can do so: **(display diagram)**

So, if it’s possible for us to create a computer-generated diagram from the problem statement, it’d be very useful.

**Proposal:**

My project will use supervised learning to perform natural-language processing on geometry problems and output a diagram corresponding to the problem statement. (To be clear, I’m going to output the Asymptote code that when compiled, will yield the diagram.)