What cybersecurity domains did each model specialize in?

For CyberBase-13B, the cybersecurity domain involves providing breakdowns of cybersecurity topics and assisting in penetration testing, which focuses on identifying weaknesses through a series of simulated attacks. For ZySec-7B, the domain supports decision-making and risk management, helping organizations restructure their security policies. For Foundation-Sec-8B, the domain is proactive threat defense, focusing on finding and understanding software vulnerabilities and weak spots. Canstralian/CyberAttackDetection, specializes in analyzing potential cyberattacks by spotting suspicious behavior and patterns in system logs.

What types of input prompts worked well (or poorly)?

For CyberBase-13B, the prompt used was to create a Python script to scan ports on a target IP. Since the task was specific with a clear instruction and topic, it worked well, showing that the model responds best when the prompt isn't vague. With ZySec-7B, the prompt asked to compare security frameworks. Since the prompt was clear and specific, the model understood the instruction and compared the frameworks and guidelines that work best. For Foundation-Sec-8B, the prompt asked the model to break down a known CVE. It performed well with structured inputs, showing it works best when referencing real security vulnerabilities. With Canstralian/CyberAttackDetection, the prompt asked the model to analyze a structured log to determine if it indicated an attack. It correctly identified threat patterns like repeated login failures, showing that detailed prompts work best. In general, the types of input prompts that work well across all models are specific and clearly written. Although each model has its own strengths, vague prompts would give generic or unclear answers.

What limitations or biases did you notice?

For CyberBase-13B, a limitation is that the code it generates can leave out necessary or outdated information. Even though it is fine tuned with a large dataset, it presents biases which affect the accuracy of the information. For ZySec-7B, although its dataset covers over 30 unique domains, there are gaps in its knowledge. This means the information it provides can be inaccurate or outdated. There are also biases present in the training data, including discriminatory assumptions in cybersecurity areas. For Foundation-Sec-8B, a limitation is that it focuses on being optimized for cybersecurity, so it may not work well in broader uses. There are geographic or cultural biases in how it recommends or explains certain security approaches. For Canstralian/CyberAttackDetection, the model is limited to processing only English network logs and structured formats. It also shows bias toward more common attack types, overlooking less frequent threats that weren't focused in its training data.

Would you trust these models for autonomous use? Why or why not?

CyberBase-13B is a base model and cannot perform well without additional support. Since its focus is writing scripts, it often generates code that is incomplete or incorrect, and may not fulfill the full requirements of the input prompt. Because of this, I would not trust it for autonomous use. Though ZySec-7B has a high rate for accuracy, it still depends on the quality of its training data. It may miss important legal requirements, which could lead to future risks. So I would not trust it to solely be autonomous. Foundation-Sec-8B is designed to make critical security decisions, so without testing and human review, I would not rely on it for autonomous use. For Canstralian/CyberAttackDetection, although it analyzes network logs effectively, it can overlook subtle threats or raise false alarms, so I would not trust it to be autonomous. Overall, I would not consider any of these models trustworthy enough for autonomous use. The models can

make mistakes, therefore, relying on them with no human involvement could result in incorrect information, overlook threats, or raise false alerts.