

## BAB V PENUTUP

### 5.1. Kesimpulan

Merujuk pada tiga rumusan masalah di Subbab 1.2, penelitian ini menjawab ketiganya, dengan catatan pada rumusan kedua. Sistem terbukti mampu menerapkan kNN dan *Logistic Regression* dalam arsitektur *streaming* berbasis Apache Flink untuk mendeteksi aktivitas tidak normal, dibuktikan oleh keberhasilan pemrosesan 20.000 *event* tanpa kegagalan infrastruktur, sekaligus menegaskan bahwa integrasi *machine learning* ke dalam *pipeline streaming* dapat diwujudkan dengan kompleksitas infrastruktur yang relatif ringan dan terjangkau untuk direplikasi pada skala laboratorium serupa. Pada rumusan kedua, capaian belum sepenuhnya sesuai ekspektasi awal. Evaluasi *batch* menunjukkan akurasi kedua model di atas 99% (kategori "Sangat Baik"), namun turun ke kategori "Baik" saat diuji pada kondisi *streaming* yang sesungguhnya. Meski demikian, latensi kedua model tetap jauh di bawah ambang batas "Sangat Baik" (1.000 ms), kNN pada 6,4243 ms dan *Logistic Regression* pada 0,4494 ms, sehingga tujuan sistem sebagai mekanisme *streaming analytics* tetap tercapai secara operasional. Latensi di bawah 8 milidetik per *event* ini menjadi nilai tawar utama dari sisi bisnis: kapabilitas deteksi ancaman yang responsif dapat diwujudkan tanpa investasi infrastruktur mahal, sehingga tetap terjangkau bagi institusi dengan anggaran teknologi informasi terbatas. Pada rumusan ketiga, tantangan utama yang teridentifikasi mencakup keterbatasan MongoDB *standalone* yang tidak mendukung *Change Streams*, serta kesenjangan performa akibat fitur agregat yang baru terakumulasi bertahap pada tahap awal *streaming*. Pilihan desain yang memprioritaskan *recall* di atas presisi turut mencerminkan prinsip kehati-hatian yang lazim diterapkan pada sistem deteksi dini keamanan siber. Dengan demikian, kontribusi penelitian ini tidak hanya bersifat akademik, tetapi juga menyediakan cetak biru teknis yang dapat dipertimbangkan unit teknologi informasi kampus untuk memperkuat postur keamanan jaringan tanpa memerlukan investasi platform *enterprise* yang lebih kompleks.

### 5.2. Saran

Berdasarkan keterbatasan yang ditemukan selama penelitian, beberapa saran diajukan untuk pengembangan sistem maupun penelitian selanjutnya. Migrasi arsitektur MongoDB dari konfigurasi *standalone* ke *replica set* disarankan sebagai prioritas utama pengembangan infrastruktur. Dengan arsitektur *replica set*, fitur *Change Streams* dapat diaktifkan sehingga akuisisi data dapat dilakukan secara *event-driven* melalui integrasi *Change Data Capture* (CDC) yang *native* dengan Flink. Pendekatan ini berpotensi menghasilkan karakteristik *true streaming* yang lebih konsisten pada keseluruhan lapisan *pipeline*, baik pada tahap ingesti maupun pemrosesan data.