

BAB VI

KESIMPULAN DAN SARAN

Bab ini berisi penjelasan mengenai kesimpulan dan saran dari penelitian yang telah dilaksanakan. Kesimpulan merupakan rangkuman hasil akhir penelitian ini berkaitan dengan tujuan yang telah ditetapkan, sedangkan saran merupakan rekomendasi yang diberikan peneliti berdasarkan hasil akhir penelitian agar dapat dijadikan bahan untuk perbaikan maupun pengembangan kedepannya.

6.1 Kesimpulan

Berdasarkan hasil analisis, perancangan, dan pengujian yang telah dilakukan pada sistem analisis topik *metadata* video Instagram untuk mendukung strategi penerbitan berita di Padang TV, maka dapat ditarik beberapa kesimpulan sebagai berikut:

1. Penerapan *pipeline* BERTopic yang mengintegrasikan model *contextual embedding* SBERT (*paraphrase-multilingual-MiniLM-L12-v2*), reduksi dimensi UMAP, dan *clustering* HDBSCAN terbukti berhasil dalam mengekstraksi topik dari *metadata* video Instagram yang memiliki panjang teks bervariasi dengan karakter bahasa beragam. Penelitian ini berhasil merumuskan 26 label interpretasi topik akhir yang relevan sesuai kebutuhan redaksi.
2. Evaluasi model BERTopic dengan nilai *coherence score* sebesar 0,5599 dan *topic diversity* sebesar 0,7278 menunjukkan hasil pengelompokan yang baik berdasarkan keterkaitan semantik dan variasi yang tinggi antartopik. Hal ini didukung oleh hasil interpretasi label yang mudah dipahami oleh tim redaksi Padang TV berdasarkan hasil UAT.
3. Integrasi algoritma *Logistic Regression* berbasis verifikasi *human-in-the-loop* (HITL) menghasilkan performa klasifikasi yang terkalibrasi dengan baik berdasarkan hasil *Brier score* sebesar 0,0231. Nilai *Brier score* tersebut menandakan bahwa dengan perolehan metrik evaluasi *accuracy* sebesar 60,27%, *precision* sebesar 0,6509, *recall* sebesar 0,62, dan *F1-score* sebesar 0,5946, model menggambarkan kinerja yang jujur di mana model kesulitan

memprediksi label topik dengan benar. Selain itu, analisis *confusion matrix* secara visual menunjukkan dominasi nilai pada garis diagonal yang mengindikasikan bahwa sebagian besar data berhasil diprediksi sesuai dengan label topik sebenarnya. Hal ini menunjukkan bahwa model mampu membedakan sebagian besar kelas topik dengan baik, meskipun masih terdapat beberapa kesalahan klasifikasi pada sejumlah topik yang memiliki karakteristik teks yang serupa.

4. Pembangunan *dashboard* interaktif berbasis *framework* Flask berhasil menjembatani kompleksitas model NLP dengan kebutuhan operasional redaksi Padang TV. Sistem ini mampu mendukung proses *agenda setting* menjadi lebih objektif berbasis data dan membantu memahami isu atau topik yang berkembang sehingga dapat mendukung strategi penerbitan berita di Padang TV.
5. Sistem berhasil menyajikan visualisasi tren topik secara kronologis yang menunjukkan dinamika isu pada konten video Instagram sesuai rentang waktu penelitian. Hal ini memungkinkan pihak redaksi Padang TV untuk memantau tren isu yang berkembang dan momentum berita sehingga berita yang dipublikasikan lebih sesuai dengan dinamika perhatian publik dan memiliki peluang lebih besar untuk menjangkau audiens.

6.2 Saran

Untuk pengembangan sistem dan penelitian lebih lanjut di masa mendatang, penulis menyarankan beberapa hal sebagai berikut:

1. Mengingat pesatnya perkembangan NLP, penggunaan model *transformer* yang lebih spesifik untuk Bahasa Indonesia formal dan non-formal dapat diuji coba untuk melihat potensinya dalam meningkatkan akurasi model lebih lanjut.
2. Untuk menjamin skalabilitas saat diakses oleh banyak pengguna secara bersamaan, pengembangan sistem disarankan menggunakan arsitektur *microservices* dan infrastruktur *cloud* yang mendukung akselerasi GPU. Hal

ini akan meningkatkan efisiensi proses inferensi data serta memudahkan pengelolaan beban kerja komputasi NLP yang berat.

3. Ruang lingkup penelitian ini terbatas hingga tahap *Construction* dalam metode *Waterfall*. Oleh karena itu, disarankan untuk melanjutkan penelitian ke tahap *Deployment* pada server produksi agar sistem dapat diakses secara *real-time*.
4. Penelitian selanjutnya disarankan untuk menerapkan skrip pengujian otomatis pada arsitektur *backend* Flask setelah sistem memasuki fase pemeliharaan jangka panjang. Pengujian *unit testing* terisolasi sulit diimplementasikan pada fase ini karena adanya fungsionalitas yang bergantung erat pada *pipeline* BERTopic, sehingga jika dipaksakan hanya akan menghasilkan validasi semu yang tidak mencerminkan kinerja model. Sementara itu, untuk *integration testing* belum diotomatisasikan karena pada penelitian ini, pengujian hubungan antar-modul telah diwakili secara manual melalui pengujian *black-box testing*.

