LAPORAN PROYEK NATURAL LANGUAGE PROCESSING (NLP)

Analisis Sentimen dan Deteksi Spam pada Komentar TikTok Institusi Pendidikan

Anggota Kelompok 4:

1. ATHA DHAIFFATIN (1003230026)
2. AZRA MEUTHIA RINALDY ((1003250051))
3. ZIKRI FIRMANSYAH (1003230043)

INSTITUT TEKNOLOGI TANGERANG SELATAN

2026

DAFTAR ISI

BAB 1: PENDAHULUAN

1.1 Latar Belakang

Media sosial, khususnya TikTok, telah menjadi sarana utama bagi institusi pendidikan untuk mempromosikan kegiatan dan berinteraksi dengan calon mahasiswa. Namun, tingginya volume interaksi seringkali membawa dua tantangan utama: sulitnya memetakan sentimen audiens secara manual dan banyaknya komentar *spam* (promosi tidak relevan) yang mengganggu informasi. Oleh karena itu, diperlukan sistem otomatis berbasis *Machine Learning* untuk menganalisis sentimen dan menyaring spam.

1.2 Tujuan

Tujuan dari proyek ini adalah:

1. Membangun aplikasi berbasis web (Flask) untuk analisis teks otomatis.
2. Mengklasifikasikan komentar menjadi sentimen Positif, Negatif, atau Netral.
3. Mendeteksi komentar yang bersifat Spam atau Bukan Spam.
4. Memvisualisasikan hasil analisis untuk pengambilan keputusan eksekutif.

BAB 2: DESKRIPSI DATA

2.1 Sumber Data Prediksi (dataNew.csv)

Data yang digunakan sebagai objek analisis (data uji/prediksi) berasal dari ekspor aktivitas media sosial TikTok yang disimpan dalam file dataNew.csv.

- **Topik:** Konten seputar institusi pendidikan (ITTS, ITB), event kampus, dan promosi beasiswa.
- **Atribut Data:** video_url, nama_akun, caption, comments, likes, shares.
- **Fokus Analisis:** Kolom comments yang berisi opini publik. Karena satu baris data bisa memiliki banyak komentar yang dipisahkan koma, dilakukan proses *explode* data untuk menganalisis per komentar.

2.2 Data Latih (Training Data)

Karena dataNew.csv adalah data mentah tanpa label, model dilatih menggunakan dataset terlabel yang didefinisikan secara internal dalam sistem (main.py).

- **Sampel Sentimen:** Berisi kalimat dengan label positif, negatif, dan netral (Contoh: "Kampus ini keren banget" [Positif], "Biaya mahal fasilitas rusak" [Negatif]).
- **Sampel Spam:** Berisi kalimat dengan label spam dan bukan_spam (Contoh: "Cek IG kita kak" [Spam], "Info pendaftaran kapan?" [Bukan Spam]).

BAB 3: PIPELINE NLP & METODOLOGI

Proses analisis dilakukan mengikuti tahapan standar NLP sebagai berikut:

3.1 Preprocessing (Pra-pemrosesan Teks)

Sebelum masuk ke model, data teks mentah dibersihkan melalui fungsi preprocess_text di main.py:

1. **Case Folding:** Mengubah semua huruf menjadi huruf kecil (*lowercase*).
2. **Cleaning:** * Menghapus URL (http/https).
    - Menghapus *mention* (@username).
    - Menghapus karakter non-alfanumerik (tanda baca).
    - Menghapus angka.
3. **Tokenization:** Memecah kalimat menjadi kata-kata individual.
4. **Stopword Removal:** Menghapus kata hubung umum bahasa Indonesia (seperti "dan", "yang", "di", "ke") menggunakan pustaka NLTK dan daftar kustom.

3.2 Feature Extraction (Ekstraksi Fitur)

Mengubah teks menjadi format numerik agar bisa diproses mesin.

- **Metode:** TF-IDF (*Term Frequency-Inverse Document Frequency*).
- **Alasan:** Metode ini memberikan bobot lebih tinggi pada kata-kata unik yang penting untuk klasifikasi, bukan hanya menghitung frekuensi kemunculan kata semata.

3.3 Model Machine Learning

Kami membandingkan tiga algoritma untuk menentukan model terbaik:

1. **Support Vector Machine (LinearSVC):** Efektif untuk ruang dimensi tinggi (teks).
2. **Naive Bayes (MultinomialNB):** Cepat dan efisien untuk klasifikasi teks standar.
3. **Decision Tree:** Model berbasis pohon keputusan yang mudah diinterpretasi.

Berdasarkan evaluasi dalam kode, **LinearSVC** dipilih sebagai model utama untuk melakukan prediksi pada data baru karena performanya yang stabil pada dataset teks kecil hingga menengah.

BAB 4: IMPLEMENTASI SISTEM

4.1 Teknologi yang Digunakan

- **Bahasa:** Python 3.x
- **Framework Web:** Flask (untuk antarmuka pengguna).
- **Library Data:** Pandas, NumPy.
- **Library NLP/ML:** NLTK, Scikit-learn.
- **Visualisasi:** Matplotlib, Seaborn.

4.2 Cara Menjalankan Aplikasi

1. Pastikan Python dan library terinstall (pip install flask pandas nltk scikit-learn matplotlib seaborn).
2. Letakkan file main.py dan dataNew.csv dalam satu folder.
3. Buka terminal/command prompt di folder tersebut.
4. Jalankan perintah:
   python main.py

5. Akses aplikasi melalui browser di alamat yang muncul (biasanya http://127.0.0.1:5000).

4.3 Fitur Aplikasi

- **Dashboard:** Menampilkan ringkasan visual (Pie chart sentimen, Bar chart spam).
- **Analisis Data:** Tabel interaktif yang menampilkan komentar asli, hasil prediksi sentimen, hasil deteksi spam, dan tingkat kepercayaan model (*confidence score*).
- **Evaluasi Model:** Halaman khusus yang membandingkan akurasi antar algoritma.
- **Uji Prediksi Manual:** Form input untuk menguji kalimat baru secara *real-time*.

BAB 5: EVALUASI DAN HASIL

5.1 Matrik Evaluasi

Model dievaluasi menggunakan teknik *Cross Validation* dengan metrik:

- **Accuracy:** Ketepatan keseluruhan prediksi.
- **Precision:** Tingkat ketepatan data positif yang diprediksi.
- **Recall:** Tingkat keberhasilan model menemukan kembali informasi.
- **F1-Score:** Rata-rata harmonik antara Precision dan Recall.

5.2 Hasil Visualisasi

Berdasarkan output dari main.py, sistem menghasilkan visualisasi berikut:

1. **Confusion Matrix:** Memetakan prediksi benar vs salah untuk melihat di mana model sering melakukan kesalahan (misal: sering salah membedakan 'netral' dan 'positif').
2. **Perbandingan Model:** Grafik batang yang menunjukkan bahwa SVM (LinearSVC) cenderung memiliki akurasi tertinggi dibandingkan Naive Bayes pada dataset ini.
3. **Distribusi Data (Pie Chart):** Menunjukkan proporsi komentar positif, negatif, dan netral dari dataNew.csv.

BAB 6: KESIMPULAN

6.1 Analisis Hasil

Dari hasil eksekusi program terhadap data dataNew.csv, dapat disimpulkan bahwa:

1. Sistem berhasil melakukan *parsing* kolom komentar yang tergabung dalam satu sel CSV menjadi baris-baris terpisah untuk dianalisis.
2. Preprocessing sangat berpengaruh dalam membersihkan *noise* seperti link promosi atau *mention* yang tidak relevan dengan sentimen.
3. Model SVM memberikan hasil klasifikasi yang cukup akurat dalam memisahkan komentar spam (promosi jualan, link afiliasi) dari komentar organik (pertanyaan seputar kampus).

6.2 Kesimpulan Akhir

Aplikasi ini memenuhi persyaratan tugas untuk melakukan pipeline NLP lengkap. Penggunaan antarmuka web mempermudah pengguna awam dalam membaca hasil analisis tanpa perlu melihat kode. Untuk pengembangan selanjutnya, dataset latih (training data) perlu diperbanyak agar model lebih mengenali bahasa gaul (*slang*) yang sering muncul di TikTok.