

# Analisis Klasifikasi Saran pada Evaluasi Pelatihan ASN Berbasis FastText Embedding dan Convolutional Neural Network

Alfian Najib Anshori

Balai Pengembangan Kompetensi PU Wilayah V Yogyakarta, Jl. Ngeksigondo 1-2, Kotagede, Yogyakarta  
corresponding author: alfian.najib@pu.go.id

---

## Abstrak/Abstract

Peningkatan kualitas pelayanan pelatihan ASN sangat bergantung pada analisis umpan balik peserta. Evaluasi numerik belum mampu mengungkap masalah spesifik, sementara analisis manual terhadap volume saran yang besar kurang efisien dan rentan subjektivitas. Pemanfaatan Natural Language Processing (NLP) dapat digunakan sebagai salah satu solusi untuk mengatasi masalah tersebut. Pemanfaatan NLP dilakukan dengan mengintegrasikan FastText sebagai *word embedding* dan arsitektur Deep Learning Convolutional Neural Network (CNN) sebagai *classifier* saran peserta ke dalam empat aspek layanan yaitu materi, pengajar, tata laksana, dan sarana prasarana. Dataset berjumlah 418 baris saran dari akun sistem informasi pengelolaan pelatihan Bapekom PU Wilayah V Yogyakarta selama tahun 2021-2024 dengan prapemrosesan dan *sampling* secara berjenjang (80% data latih, 20% data uji). Hasil pelatihan menunjukkan *training accuracy* sebesar 90,28%, *validation accuracy* sebesar 80,08% dan *test accuracy* keseluruhan sebesar 73,0%. Model menunjukkan kemampuan klasifikasi dasar yang memadai, meskipun terdapat indikasi *overfitting*. Analisis metrik rinci mengungkap kinerja yang bervariasi antar kelas dan misklasifikasi yang terjadi pada kelas dengan variasi kosakata yang cukup besar. Kinerja model cukup baik dan dapat berfungsi sebagai *baseline*, serta membuka peluang pengembangan lebih lanjut dalam otomasi analisis umpan balik di bidang pelatihan ASN. Hasil klasifikasi dapat menjadi alat bantu pendukung pengambilan keputusan manajemen dalam mengidentifikasi area yang perlu segera ditangani.

*This study addresses the limitations of conventional numerical evaluation and manual review in processing participant feedback from civil servant (ASN) training programs by developing text-classification model. A Natural Language Processing could be used as a solution. This approach was implemented by integrating FastText word embeddings which is capable of representing out of vocabulary items via subword modeling and a Convolutional Neural Network (CNN) for efficient semantic pattern extraction. The empirical dataset comprises 418 suggestion entries collected from the training management information system of Bapekom PU Region V Yogyakarta for the period 2021–2025. After preprocessing, the dataset was partitioned stratified into 80% training and 20% testing subsets. Experimental results indicate a training accuracy of 90,28%, validation accuracy of 80,08%, and an overall test accuracy of 73,0% (N = 126), suggesting adequate baseline classification performance alongside indications of overfitting. Class-wise analysis reveals heterogeneous performance and recurring misclassifications for categories characterized by high lexical variability. The model's performance is deemed adequate, establishing a solid baseline that presents a viable opportunity for further development in the automation of feedback analysis for civil servant (ASN) training. The classification results can serve as a decision-support tool for management to efficiently identify areas requiring immediate attention.*

This is an open access article under the CC–BY-SA license.



---

**Kata Kunci:** FastText, CNN, Klasifikasi, Saran, Pelatihan  
**Keywords:** FastText, CNN, Classification, Feedback, Training

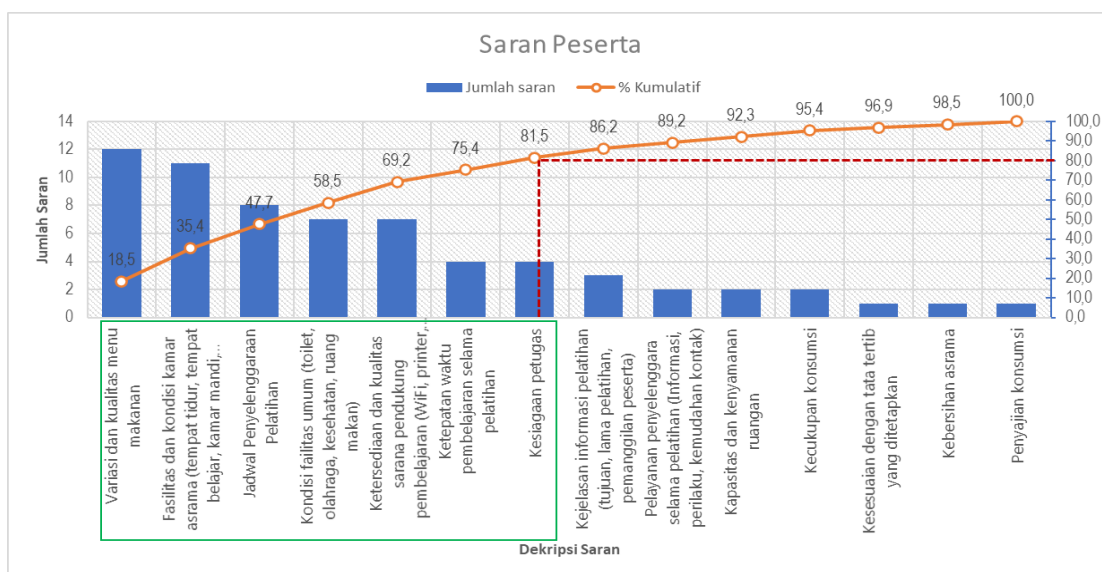
---

# 1. Pendahuluan

Pengembangan kompetensi Aparatur Sipil Negara (ASN) melalui program pelatihan merupakan salah satu instrumen penting dalam meningkatkan kinerja lembaga negara. Penyelenggaraan pelatihan berada di bawah koordinasi unit organisasi yang menangani Sumber Daya Manusia (SDM) di dalam Kementerian/Lembaga. Keberhasilan penyelenggaraan pelatihan secara umum diindikasikan oleh tiga aspek utama pelayanan, yaitu: (1) materi, (2) pengajar, dan (3) sarana dan prasarana pelatihan.

Hasil evaluasi penyelenggaraan pelatihan merupakan salah satu faktor penting dalam peningkatan layanan lembaga pelatihan. BPSDM Kementerian Pekerjaan Umum (PU) sebagai salah satu lembaga penyelenggara pelatihan mendasarkan evaluasi pelatihan didasarkan pada aspek materi, pengajar, tata laksana, dan sarana prasarana pelatihan, dimana kewenangan Balai Pengembangan Kompetensi sebagai Unit Kerja Pelaksana Pelatihan berada pada ranah tata laksana dan sarana prasarana pelatihan. Evaluasi pelaksanaan pelatihan di BPSDM PU didukung dengan sistem informasi pengelolaan pelatihan terintegrasi Sibangkoman.

Output utama hasil evaluasi adalah berupa indeks kepuasan numerik pada skala 1 (sangat buruk) hingga 6 (sangat baik), serta kolom isian saran. Meskipun indeks numerik memberikan gambaran umum capaian layanan, tetapi belum mampu mengungkap permasalahan spesifik yang tersembunyi. Sebagai contoh adalah keluhan terkait dengan penerangan ruangan atau sambungan internet yang lolos dari deteksi panitia, dan tersampaikan melalui catatan teks setelah pelatihan selesai. Panitia harus melakukan pemilahan terhadap saran-saran yang masuk untuk dianalisis sebagai bahan peningkatan kualitas layanan. Gambar 1 menunjukkan hasil pengolahan secara manual terhadap saran peserta pelatihan di Balai Pengembangan Kompetensi PU Wilayah V Yogyakarta selama tahun 2021 hingga 2023 menggunakan metode Pareto. Hasil tersebut menunjukkan hal-hal yang perlu menjadi perhatian manajemen untuk segera ditindaklanjuti seperti variasi makan, jadwal, fasilitas, dan aspek-aspek lain



Gambar 1. Hasil analisis Pareto

Sumber: hasil analisis (2022)

Proses manual dilakukan dengan mengamati setiap umpan balik peserta untuk kemudian diklasifikasikan ke dalam kategori tata laksana, layanan petugas, konsumsi, atau sarana prasarana. Metode tersebut dilakukan pula oleh Sari, U. C., & Ulfiana, D. (2021) atau Yojana (2023) untuk mendapatkan kesimpulan umum dari hasil evaluasi. Firdausy et al. (2023) menyatakan bahwa penelitian terkait dengan reaksi kualitatif peserta pelatihan relatif jarang dilakukan, sehingga pengembangan metode masih sangat memungkinkan untuk dilakukan. Ketika dilakukan dalam jumlah yang relatif kecil cara manual cukup efektif. Namun apabila volume saran peserta dapat sangat besar dalam rentang waktu tertentu sehingga analisis secara manual menjadi tidak efisien dan rawan subjektivitas. Oleh karena itu, pendekatan *text mining* dan/atau *machine learning* menjadi solusi terkini yang potensial untuk melakukan analisis terhadap saran peserta pelatihan.

*Natural Language Processing (NLP)* merupakan cabang dari kecerdasan artifisial yang fokus pada pemrosesan dan pemahaman bahasa alami oleh komputer. Seiring meningkatnya kompleksitas data tekstual dan kebutuhan terhadap analisis otomatis dalam berbagai sektor, pendekatan tradisional berbasis aturan (*rule-based systems*) dan statistik dalam NLP mulai ditinggalkan karena keterbatasannya dalam skala pemrosesan dan kemampuan generalisasi. Sebagai respons terhadap tantangan tersebut, pendekatan berbasis *Machine Learning (ML)* menjadi landasan utama dalam pengembangan sistem NLP modern.

Dalam konteks NLP, *Machine Learning* berperan sebagai kerangka kerja adaptif yang memungkinkan model untuk mempelajari data tekstual dan melakukan prediksi berdasarkan pola yang diekstraksi selama proses pelatihan. Tugas-tugas klasifikasi teks, analisis sentimen, dan pemodelan topik kini banyak diselesaikan dengan algoritma ML seperti *Naïve Bayes*, *Support Vector Machine (SVM)*, hingga model berbasis jaringan saraf seperti *Recurrent Neural Networks (RNN)* dan *Convolutional Neural Networks (CNN)*.

*Machine Learning* menjadi penguat NLP dalam membangun sistem yang mampu memahami bahasa alami secara lebih akurat dan fleksibel. Secara umum, model klasifikasi teks tradisional (*Naïve Bayes*, SVM, RNN) memerlukan ekstraksi fitur manual yang memakan waktu, dan proses penyesuaian *hyperparameter* yang kompleks. Sebaliknya, algoritma *deep learning* (CNN, BERT, atau Transformer) mampu melakukan *feature learning* secara otomatis, meskipun memerlukan jumlah data yang jauh lebih besar (Li et al, 2022).

Berbagai penelitian empiris selama lima tahun terakhir telah membuktikan efektivitas *Convolutional Neural Network (CNN)* dalam klasifikasi teks dan analisis sentimen pada korpus berbahasa Indonesia. Listyarini dan Anggoro (2021) menunjukkan bahwa arsitektur CNN empat lapis mampu mencapai akurasi 90 % dalam tugas analisis sentimen terhadap 500 sampel *tweet* Pilkada 2020, sedangkan Bachri dan Gunawan (2024) melaporkan akurasi mencapai 99,6 % pada tugas deteksi *spam email bilingual* Inggris–Indonesia. Airlangga (2024) menambahkan bukti dengan menguji dataset SMS *spam*, yang mencatat akurasi mendekati 99 %, menegaskan bahwa CNN, baik dalam bentuk tunggal maupun kombinasi dengan jaringan RNN, unggul dalam mengekstraksi pola lokal dan spasial dari teks.

Sementara FastText sebagai perangkat *word embedding* telah diadopsi secara luas sebagai *pretrained embedding* yang unggul dalam menangani kata-kata di luar kosakata melalui pendekatan *subword*. Illahi dan Setiawan (2024) melaporkan peningkatan akurasi model RNN dari 79,32 % menjadi 82,72 %

setelah menyertakan fitur *FastText* pada analisis *tweet* Pemilu 2024. Purnamadewi & Zahra (2024) secara khusus menguji kombinasi FastText–CNN untuk mendeteksi email *spam* dengan akurasi dan presisi 98,4%. Secara umum, bukti empiris tersebut menunjukkan bahwa integrasi FastText, CNN dan integrasi keduanya dapat dimanfaatkan dalam tugas klasifikasi teks yang andal dan aplikatif.

Namun demikian, penelitian-penelitian tersebut secara umum belum mengakomodir *multiclass*, dan lebih spesifik digunakan pada bidang penyelenggaraan pelatihan ASN. Oleh karena itu, penelitian bertujuan merancang model dasar memanfaatkan kelebihan CNN dan *FastText* untuk dengan penekanan pada domain pelatihan birokrasi dan kosakata spesifik instansi pemerintahan.

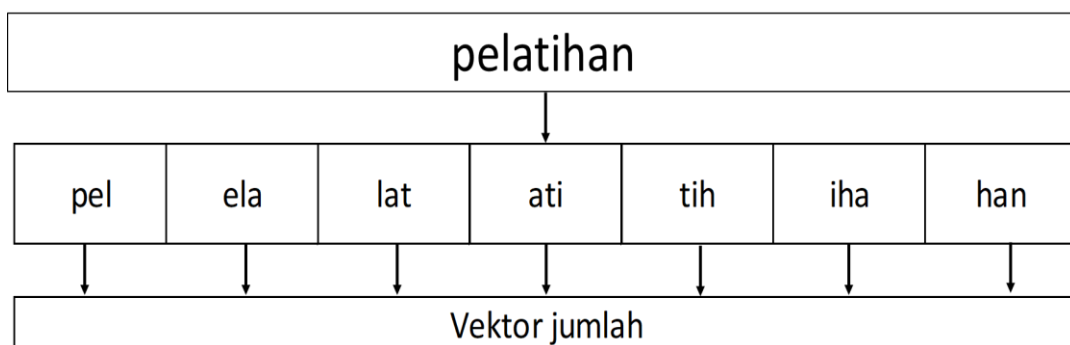
Penelitian ini berupaya menyusun model untuk mengklasifikasi saran peserta pelatihan berdasarkan aspek penyelenggaraan pelatihan (materi, pengajar, maupun sarana dan prasarana). Hasil penelitian secara praktis dimanfaatkan untuk mendukung efektifitas evaluasi pelaksanaan pelatihan di Bapekom PU Wilayah V Yogyakarta. Model dasar yang dihasilkan dapat membantu panitia dan manajemen untuk mengidentifikasi aspek yang perlu ditangani, sehingga meningkatkan kualitas layanan pelatihan secara berkelanjutan.

Berdasarkan uraian latar belakang, pertanyaan yang dapat diajukan adalah Bagaimana proses merancang model klasifikasi teks yang memanfaatkan *FastText Embeddings* dan *Convolutional Neural Network* untuk mengkategorikan saran peserta pelatihan?

## 2. Metodologi

### 2.1. *FastText Embedding*

FastText merupakan salah satu pendekatan representasi kata yang dikembangkan untuk mengatasi keterbatasan model *embedding* tradisional (misalnya Word2Vec) dalam menangani kata-kata di luar kosakata *umum* (*out-of-vocabulary/OOV*). Gambar 2 menunjukkan ilustrasi *word embedding* pada model *FastText*.



Gambar 2. Ilustrasi *word embedding* pada *FastText*

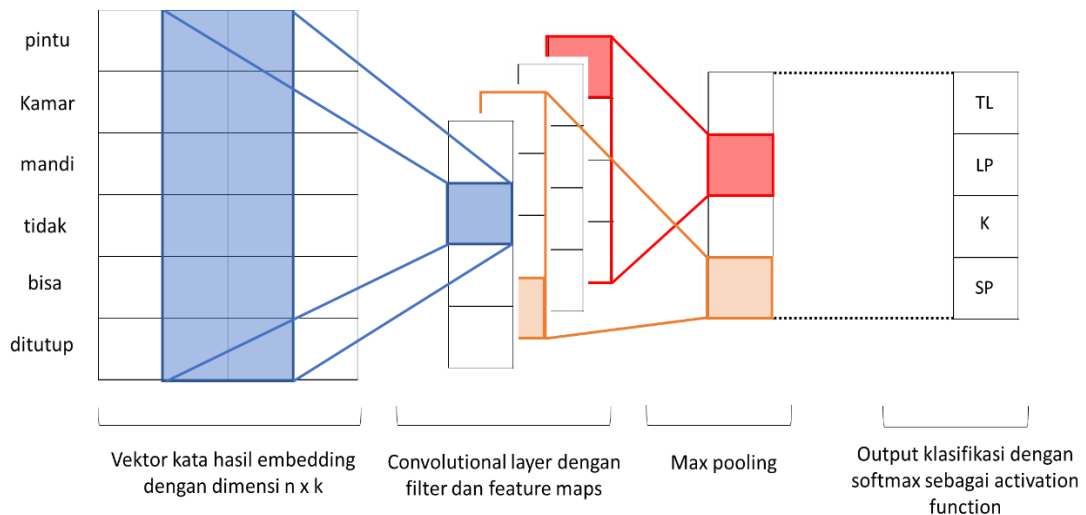
Metode *embedding* tersebut memodelkan kata dengan memecah kata dalam bentuk himpunan *n-gram* karakter, sehingga memungkinkan representasi yang lebih fleksibel dan handal terhadap kata baru atau susunan kata yang kompleks (Bojanowski et al., 2017). Pendekatan tersebut tidak hanya menghasilkan representasi kata yang lebih akurat, tetapi turut mempercepat proses pelatihan model.

Dalam konteks Bahasa Indonesia, penerapan FastText menunjukkan performa yang baik. Penelitian Illahi dan Setiawan (2024) menunjukkan bahwa

penambahan *embedding* FastText ke dalam model RNN pada analisis sentimen *tweet* Pemilu 2024 meningkatkan akurasi klasifikasi sebesar 3,4 %. Keunggulan utama FastText terletak pada kemampuan dalam merepresentasikan kata-kata baru melalui *subword*, proses komputasi kecepatan tinggi, serta fleksibilitas dalam integrasi ke berbagai *data pipeline* untuk klasifikasi.

## 2.2. Pemanfaatan *Convolutional Neural Network (CNN)* untuk Klasifikasi Teks

*Convolutional Neural Network (CNN)* secara umum digunakan dalam pemrosesan citra (*images*). Namun sejak Kim (2014) memperkenalkan penggunaan dalam NLP, pemanfaatan CNN terbukti efektif dalam tugas klasifikasi teks. Arsitektur CNN dalam pemrosesan teks bekerja dengan menerapkan operasi konvolusi terhadap urutan vektor *embedding* kata untuk mengekstraksi fitur lokal berupa pola *n-gram*. CNN memiliki sejumlah keunggulan, antara lain kemampuan pembelajaran fitur otomatis, *pooling*, serta efisiensi komputasi yang lebih tinggi dibandingkan model berbasis *Recurrent Neural Network (RNN)*. Gambar 3 menunjukkan ilustrasi arsitektur tipikal CNN yang telah disesuaikan dengan penelitian ini.



Gambar 3. Ilustrasi arsitektur CNN

Beberapa penelitian menunjukkan bahwa CNN bekerja secara optimal pada teks pendek maupun panjang. Penelitian Listyarini dan Anggoro (2021) menghasilkan akurasi sebesar 90% pada tugas analisis sentimen tweet Pilkada 2020 menggunakan CNN empat lapis. Bachri dan Gunawan (2024) mencatat akurasi mencapai 99,6% dalam mendeteksi surel spam dwibahasa menggunakan arsitektur CNN.

Ramadhan dan Setiawan (2023) melaporkan bahwa penerapan CNN dengan pada 17.247 tweet ulasan film meningkatkan akurasi dan F1-Score di tiga aspek (alur, akting, sutradara) dibanding *baseline* tradisional. Ranti dan Girsang (2020) menemukan CNN mampu mendeteksi sarkasme pada 1.138 komentar dengan F1-Score 90%. Hasil-hasil tersebut menegaskan kemampuan CNN menangkap pola bahasa halus seperti keberadaan majas ironi. Sementara itu, Ramdhani et al. (2020) menunjukkan bahwa CNN dengan 300 *hidden layers* yang dilatih pada 377 data pelatihan dan 95 data pengujian, meraih akurasi 90,74% dan loss 29,05% pada klasifikasi berita Indonesia, menunjukkan stabilitas kinerja CNN pada dataset yang relatif kecil. Berbagai hasil tersebut menegaskan bahwa CNN mampu mengidentifikasi fitur semantik penting dalam teks secara efisien dan akurat dengan data yang terbatas.

### 2.3. Integrasi FastText dan CNN dalam Klasifikasi Teks

Integrasi antara *embedding* FastText dan arsitektur CNN telah menunjukkan hasil yang menjanjikan dalam berbagai eksperimen klasifikasi teks. FastText berperan dalam menyajikan representasi kata yang lebih kaya dan responsif terhadap kata-kata baru, sedangkan CNN bertindak sebagai ekstraktor fitur yang efisien dan akurat. Umer et al. (2023) menunjukkan bahwa kombinasi *embedding* FastText ke dalam CNN tiga lapis mampu meningkatkan akurasi klasifikasi hingga 20% dibandingkan metode tradisional. Shumaly et al. (2021) memanfaatkan FastText bersama CNN untuk analisis sentimen ulasan produk berbahasa Persia, memperoleh F1-score sebesar 0,956.

Penelitian serupa dalam Bahasa Indonesia dilakukan oleh Maulana et al. (2025) yang mengkombinasikan Fasttext dan CNN dengan akurasi 80,79%. Penelitian lain dilakukan oleh Purnamadewi dan Zahra (2024) untuk mendeteksi *phishing* melalui surel dengan akurasi dan presisi 98,4%. Hal tersebut mengindikasikan bahwa integrasi FastText dan CNN dapat membentuk prosedur klasifikasi teks yang handal dan dapat diadaptasi di berbagai domain aplikasi.

### 2.4. Penggunaan variabel -variabel evaluasi sebagai *multiclass* dalam model

Variabel-variabel evaluasi pelatihan menggunakan hasil penelitian Anshori (2024) yang membagi aspek layanan pelatihan ke dalam 4 kategori yaitu tata laksana, layanan petugas, sarana prasarana, dan konsumsi.

### 2.5. Sumber data

Data saran peserta diperoleh dari keluaran akun *administrator* sistem informasi pengelolaan pelatihan Sibangkoman Bapekom PU Wilayah V Yogyakarta dalam rentang waktu 2021–2024 sebanyak 418 baris data yang digunakan sebagai data model. Pengumpulan data dilakukan dengan menyalin secara salin – tempel data pada tabel saran peserta pada akun *administrator* Sibangkoman. Pelabelan dilakukan secara manual terhadap seluruh baris data yang mengkategorikan umpan balik peserta ke dalam kategori tata laksana, layanan petugas, konsumsi, dan sarana prasarana.

### 2.6. Adopsi Model Dasar

Penyusunan model klasifikasi dimulai dari proses prapemrosesan data teks (*text preprocessing*) meliputi pembersihan, normalisasi, *stemming*, dan tokenisasi. Seluruh dataset kemudian dibagi secara *stratified* menjadi 80% data *training* termasuk data untuk validasi, dan 20% data *testing* untuk memastikan distribusi label yang seimbang pada kedua himpunan data.

Kemudian *embedding* FastText diterapkan untuk mengonversi setiap token menjadi vektor densitas tinggi yang merepresentasikan informasi semantik dan morfologi kata. Vektor-vektor kata kemudian menjadi *input* bagi *neural network* dalam tahap pelatihan. Pada tahap tersebut, arsitektur seperti CNN digunakan untuk mengekstraksi pola spasial dan fitur dari representasi kata dalam dokumen, dengan lapisan konvolusi dan *pooling* yang dioptimalkan melalui mekanisme *backpropagation*.

Arsitektur model CNN pada penelitian ini mengadopsi hasil penelitian Purnamadewi dan Zahra (2024) yang menggunakan kombinasi Fasttext dan CNN dengan 2 lapis konvolusi.

Perbedaan pokok terletak pada *output* akhir dari model yaitu 4 kelas berbeda sebagai wujud dari kategori aspek evaluasi pelatihan, sementara Purnamadewi dan Zahra (2024) menggunakan 2 kelas sebagai kategori surel *spam* dan *non-spam*. Perbedaan tersebut menyebabkan penggunaan fungsi aktivasi yang berbeda, dimana untuk 4 kelas termasuk kategori *multiclass* sehingga

menggunakan *softmax* sebagai fungsi aktivasi. Sedangkan Purnamadewi dan Zahra (2024) menggunakan fungsi aktivasi sigmoid untuk klasifikasi biner.

## 2.7. Metrik Evaluasi Model

Model yang telah dibangun kemudian dievaluasi menggunakan metrik berikut:

1. Akurasi, yaitu proporsi prediksi data benar terhadap total sampel
2. Presisi, yaitu proporsi prediksi positif yang benar dari semua prediksi positif yang dibuat oleh model.  
$$\text{Presisi} = (\text{True Positives}) / (\text{True Positives} + \text{False Positives})$$
3. *Recall*, yaitu proporsi prediksi positif yang benar dari semua kasus positif yang terdapat dalam dataset.  
$$\text{Recall} = (\text{True Positives}) / (\text{True Positives} + \text{False Negatives})$$
4. *F1-score* adalah rata-rata harmonis dari Presisi dan Rekal. Nilai F1 dihitung dalam dua agregasi yaitu *macro-average* (rata-rata sederhana antar kelas) dan *weighted-average* (rata-rata berbobot menurut dukungan tiap kelas).
5. *Confusion Matrix* adalah perbandingan data hasil prediksi dengan data aktual, dengan rincian berikut:
  - *True Positives (TP)*: model memprediksi positif dengan hasil positif.
  - *False Positives (FP)*: model memprediksi positif, tetapi data asli menunjukkan negatif.
  - *False Negatives (FN)*: memprediksi negatif, tetapi data asli menunjukkan positif.
  - *True Negatives (TN)*: Kasus di mana model memprediksi negatif dan data asli menunjukkan negatif.

Seluruh proses dilakukan menggunakan bantuan perangkat Google Collaboratory dengan Bahasa Pemrograman Python.

## 3. Hasil dan Pembahasan

### 3.1 Penyusunan dan Pelatihan Model Dasar

Tahap awal penyusunan model dasar adalah pemrosesan data teks. Tahap tersebut merupakan proses fundamental dalam pemodelan *Natural Language Processing (NLP)* untuk memastikan kualitas dan konsistensi data. Proses ini dimulai dengan pembersihan data (*data cleaning*) yang bertujuan untuk menghilangkan karakter-karakter yang tidak relevan, seperti tautan URL, symbol atau karakter-karakter khusus (+\_\*&~%\$#@!~), angka (1234567890), dan emotikon, yang tidak memiliki kontribusi semantik terhadap proses klasifikasi.

Kemudian dilanjutkan dengan proses normalisasi untuk menyeragamkan format teks menjadi huruf kecil (*lowercase*). Proses ini memastikan kata yang sama tidak diperlakukan sebagai teks yang berbeda oleh model, sehingga dapat meningkatkan proses komputasi.

Prapemrosesan teks berikutnya adalah tokenisasi, yaitu pemecahan kalimat menjadi kata atau unit yang lebih kecil (token). Token-token tersebut menjadi satuan terkecil yang akan diproses pada tahapan *stemming*.

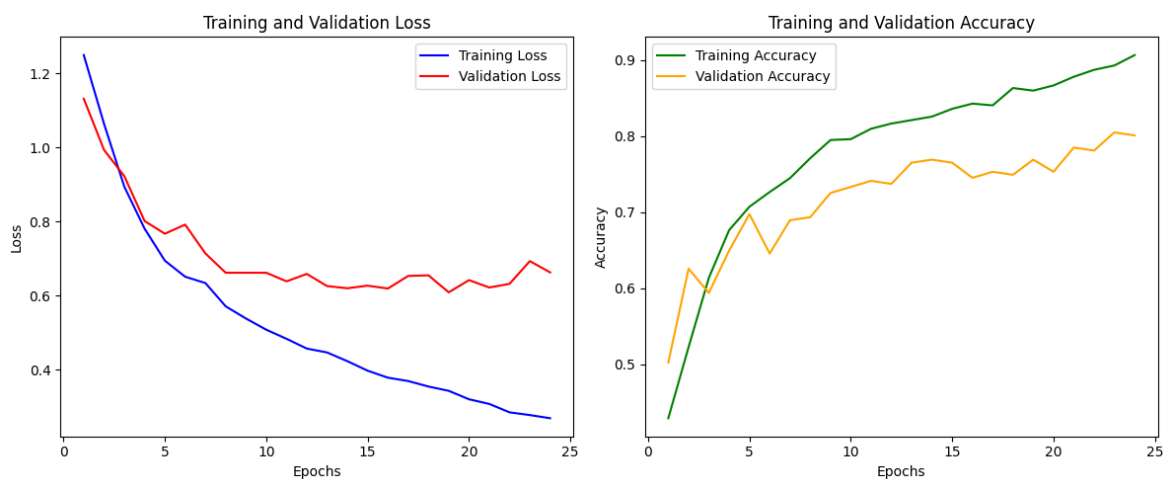
Proses *stemming* merupakan proses perubahan setiap kata menjadi menjadi bentuk atau kata dasar dengan menghapus tambahan awalan dan/atau akhiran. *Stemming* akan mengelompokkan kata-kata dasar yang sama, sebagai contoh kata 'menggunakan', 'digunakan', dan 'pengguna' menjadi kata 'guna'. Proses tersebut secara efektif mengurangi dimensi fitur dan meningkatkan kemampuan model dalam membaca pola dari dataset.

Hasil prapemrosesan teks kemudian melalui proses *word embedding* oleh model *FastText*. Proses tersebut mengubah kata menjadi sub-kata (*subword*) untuk selanjutnya ditransformasikan menjadi vektor kata. Wujud vektor kata adalah berupa matriks angka yang hanya dipahami oleh mesin atau komputer. Vektor atau matriks tersebut menjadi bahan input bagi model CNN untuk melakukan pelatihan dan pengujian. Rancangan model CNN dalam penelitian ini adalah sebagai berikut:

- *Input* : *number of tokens: 60, dimensions: FastText custom dimensions.*
- *Convolutional 1* : *filter: 32, kernel: 5, Stride:1, activation: ReLu.*
- *Max-Pooling 1* : *pool size: 2.*
- *Convolutional 2* : *filter: 64, kernel: 2, stride:1, activation: ReLu.*
- *Max-Pooling 2* : *spatial size: 2.*
- *Flatten*
- *Dense: activation: ReLu.*
- *Output* : *activation: softmax, neuron: 4 neuron for binary classification.*

Jumlah 418 baris data tergolong sedikit untuk melatih model CNN, sehingga penelitian ini melakukan data augmentation menjadi 3 kali lipat jumlah awal. *Data augmentation* merupakan mekanisme untuk mendapatkan jumlah data yang memenuhi kuantitas wajar dalam melatih model *Deep Learning*. Prosedur tersebut hanya menambah jumlah data, tetapi tidak merubah pola atau properti lain dari dataset.

Hasil pelatihan dan pengujian model menunjukkan adanya perbedaan signifikan antara kinerja model pada data latih dan data uji yang mengindikasikan fenomena *overfitting*. Gambar 4 menunjukkan indikasi *overfitting* pada model. *Overfitting* adalah kondisi ketika model menghafal data pelatihan, tetapi tidak mempelajari pola yang mendasarinya.



Gambar 4. Hasil pelatihan model

Sumber : hasil *running model*

Model mencapai akurasi pelatihan 90,28% dengan *loss* 0,2627, tetapi mengalami penurunan signifikan ketika memproses data validasi (80,08% dengan *loss* 0,6628) dan data uji independen (73% dari 126 sampel). Gambar 1 turut memperkuat indikasi tersebut dengan *training loss* terus menurun dan *training accuracy* meningkat stabil, sementara *validation loss* mulai fluktuatif setelah epoch ke-5 dan mulai berfluktuasi, diikuti *validation accuracy* yang stagnan sejak epoch ke-10. Kesenjangan antara kurva pelatihan dan validasi yang terus melebar antara garis *training* (biru dan hijau) dan garis *validation*

(merah dan oranye) menegaskan bahwa model lebih banyak “menghafal” data latih daripada mempelajari pola yang dapat digeneralisasi.

Fenomena tersebut dapat disebabkan oleh ketidakseimbangan arsitektur model dengan ketersediaan data, serta distribusi data antar kelas yang tidak seimbang sehingga model gagal menangkap variasi yang representatif. Penerapan *early stopping* yang secara umum menjadi langkah mitigasi untuk memperbaiki kinerja model telah dilakukan dengan hasil belum optimal. Potensi penyebab lain adalah ketidakseimbangan untuk masing-masing kelas data. Namun hal tersebut tidak dilakukan dalam rangka menjaga proporsi data asli mengingat keterbatasan jumlah data.

### 3.2. Evaluasi Model Dasar

Analisis metrik evaluasi pada Gambar 5 secara rinci menunjukkan ketidakseimbangan performa antar kelas (*class*). Pada kelas tata laksana, model menunjukkan presisi tinggi (0,79) tetapi *recall* rendah (0,64). Hasil tersebut mengesankan perilaku model lebih fokus pada ketepatan daripada kelengkapan, sehingga banyak nilai terlewat (*false negatives*). Namun pada kelas layanan petugas, model cenderung lebih agresif, ditunjukkan oleh *recall* tinggi (0,79) tetapi dengan presisi yang lebih rendah (0,67), yang berakibat pada peningkatan *false positives*. Kinerja stabil tercatat pada kelas sarana dan prasarana, dengan *F1-score* 0,74.

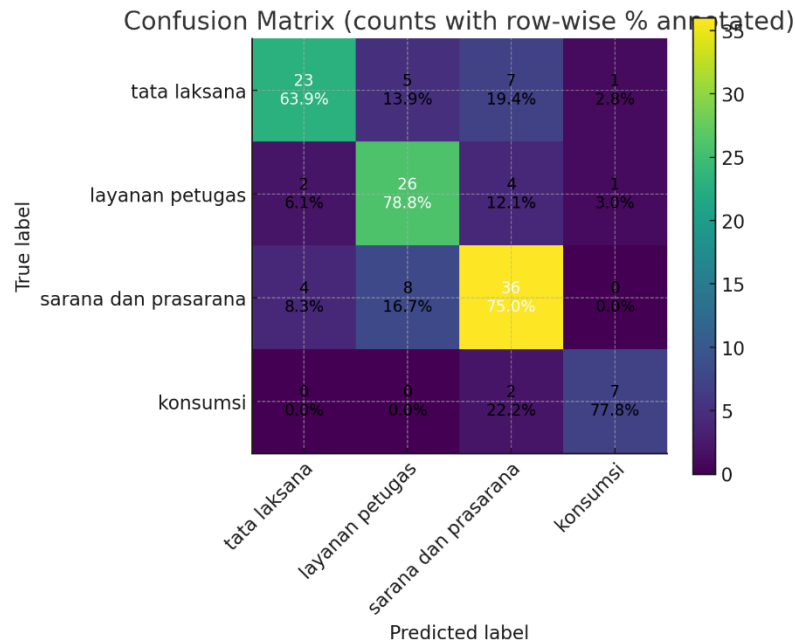
	precision	recall	f1-score	support
tata laksana	0.79	0.64	0.71	36
layanan petugas	0.67	0.79	0.72	33
sarana dan prasarana	0.73	0.75	0.74	48
konsumsi	0.78	0.78	0.78	9
accuracy			0.73	126
macro avg	0.74	0.74	0.74	126
weighted avg	0.74	0.73	0.73	126

Gambar 5. Tangkapan layar evaluasi model

Sumber: output model

Sedangkan metrik pada kelas konsumsi (*F1-score*=0,78), meskipun lebih tinggi dari tiga metrik lain, harus diinterpretasikan dengan hati-hati karena memiliki jumlah data yang sangat kecil ( $n=9$ ).

Gambar 6 merupakan *Confusion matrix* yang menunjukkan pola kesalahan yang sistematis dan menjelaskan keberagaman kinerja antar kelas. Berdasarkan keseluruhan sampel data *tata laksana*, 63,9% diklasifikasikan dengan benar, sementara 19,4% terklasifikasi sebagai sarana dan prasarana dan 13,9% sebagai layanan petugas. Kelas layanan petugas berhasil diklasifikasikan dengan benar sebesar 78,8% dari keseluruhan data, tetapi memiliki sejumlah *false positive* dari kelas lain. Kelas sarana dan prasarana tercatat relatif stabil 75,0% prediksi benar, tetapi mayoritas klasifikasi yang salah, tercatat sebagai layanan petugas (16,7%). Kelas konsumsi menunjukkan akurasi tinggi 77,8%, tetapi dengan jumlah 9 data sangat rentan terhadap misinterpretasi.



Gambar 6. *Confusion Matrix* hasil pelatihan model  
Sumber: output model

Pola tersebut mengindikasikan dua masalah yaitu, pertama, banyak teks pada dataset menggunakan kata-kata yang sama untuk beberapa kategori sehingga model kesulitan membedakan. Kedua, beberapa data mengandung lebih dari satu isu yang dapat terklasifikasi ke dalam banyak kelas, tetapi model dilatih untuk memilih satu kelas dominan sehingga dipaksa untuk memilih satu kategori. Kondisi tersebut merupakan dampak dari penggunaan fungsi aktivasi *softmax* pada arsitektur CNN. Hal tersebut menyebabkan model cenderung melewati beberapa masalah tata laksana (*false negatives*) dan memberikan terlalu banyak prediksi untuk layanan petugas (*false positives*).

Secara keseluruhan, akurasi 73% menunjukkan kemampuan yang cukup baik. Hasil tersebut lebih rendah apabila dibandingkan dengan Maulana et al. (2025) dan Purnamadewi dan Zahra (2024). Perbedaan terletak pada *output* akhir klasifikasi yaitu jumlah kelas yang digunakan. Penggunaan empat kelas berbeda pada penelitian ini membutuhkan kinerja model yang lebih baik. Peningkatan jumlah *layer* dapat menjadi salah satu alternatif apabila merujuk pada hasil Ramdhani et al. (2020) yang menggunakan 300 *hidden layer*, tapi dengan data yang lebih sedikit. Alternatif lain adalah mengurangi jumlah *layer* konvolusi menjadi satu *layer* karena data teks yang diproses memiliki kosakata cukup sederhana, sehingga mengurangi kompleksitas proses dalam model. Berdasarkan hasil analisis, peningkatan kinerja model dapat dilakukan dengan penambahan data, *hyperparameter tuning*, modifikasi *layer* konvolusi, dan penyimbangan data antar kelas (*data balancing*).

## 4. Kesimpulan & Saran

### 4.1. Kesimpulan

Penelitian bertujuan menyusun model klasifikasi teks yang memanfaatkan *FastText Embeddings* dan *Convolutional Neural Network* (CNN) untuk kategorisasi saran peserta pelatihan. Akurasi keseluruhan sebesar 73% menunjukkan model memiliki performa yang cukup baik, meskipun belum optimal dan belum

dapat digunakan untuk keperluan generalisasi data baru. Model membutuhkan lebih banyak data dan kondisi data yang lebih seimbang antar kelas. Meskipun cukup efektif digunakan untuk kasus spesifik, penerapan CNN tetap membutuhkan jumlah data nyata yang mencukupi, dan keseimbangan antar kelas yang memadai.

Model dasar hasil penelitian ini dapat digunakan sebagai alat bantu pendukung keputusan dan sebagai model dasar untuk pengembangan di masa depan. Model awal tersebut berpotensi untuk dikembangkan sebagai alat bantu otomasi dalam pengklasifikasian saran peserta pelatihan.

## 4.2. Rekomendasi

Penelitian selanjutnya dapat dilakukan dengan jumlah data nyata yang lebih besar dan kelas yang lebih seimbang. Hal lain yang dapat dilakukan adalah dengan memodifikasi jumlah *layer* konvolusi menjadi 1 *layer*, mengingat sifat data pada kasus spesifik seperti umpan balik peserta pelatihan memiliki kosakata yang lebih sederhana.

## Ucapan Terimakasih

Terima kasih penulis ucapkan kepada Kepala Balai Pengembangan Kompetensi Pekerjaan Umum Wilayah V Yogyakarta yang telah memberikan dukungan untuk akses data evaluasi pelatihan sehingga penelitian dan penulisan artikel dapat terlaksana dengan baik.

## Referensi

- Airlangga, G. 2024. A Comparative Analysis of Deep Learning Models for SMS Spam Detection: CNN-LSTM, CNN-GRU, and ResNet Approaches. *Journal of Computer Networks, Architecture and High Performance Computing*, 6(4), 1952–1960. <https://doi.org/10.47709/cnahpc.v6i4.4827>
- Anshori, A. N. 2024. Analisis Klasifikasi Saran Peserta Pelatihan Menggunakan Pendekatan Machine Learning. *Jurnal Kewidyaiswaraan* 9(2). <https://doi.org/10.56971/jwi.v9i2.323>
- Bachri, C. M., & Gunawan, W. 2024. Deteksi Email Spam menggunakan Algoritma Convolutional Neural Network (CNN). *JEPIN (Jurnal Edukasi dan Penelitian Informatika)*, 10(1), 88-94.
- Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. 2017. Enriching word vectors with subword information. *Transactions of the association for computational linguistics*, 5, 135-146.
- Firdausy, N., Yuadi, I., & Puspitasari, I. 2023. Analisis Sentimen Evaluasi Reaksi E-Learning Menggunakan Algoritma Naïve Bayes Support Vector Machine dan Deep Learning. *Techno. Com*, 22(3).
- Illahi, I. R., & Setiawan, E. B. 2024. Sentiment Analysis on Social Media Using Fasttext Feature Expansion and Recurrent Neural Network (RNN) with Genetic Algorithm Optimization. *International Journal on Information and Communication Technology (IJoICT)*, 10(1), 78-89.
- Kim, Y. 2014. Convolutional Neural Networks for Sentence Classification. In A. Moschitti, B. Pang, & W. Daelemans (Eds.), *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 1746–1751). Association for Computational Linguistics. <https://aclanthology.org/D14-1181>.
- Li, Q., Peng, H., Li, J., Xia, C., Yang, R., Sun, L., Yu, P. S., & He, L. 2022. A Survey on Text Classification: From Traditional to Deep Learning. *ACM Transactions on Intelligent Systems and Technology*. <https://doi.org/10.1145/3495162>.

- Maulana, M. T., Muflikhah, L., & Fatyanosa, T. N. 2025. Analisis Sentimen Pengguna Indodax Menggunakan FastText dan Convolutional Neural Network (CNN). *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 9(6).
- Purnamadewi, Y. R., & Zahra, A. 2025. Enhancing Detection of Zero-day Phishing Email Attacks in The Indonesian Language using Deep Learning Algorithms. *Bulletin of Electrical Engineering and Informatics*, 14(1), 505–512. <https://doi.org/10.11591/eei.v14i1.8759>.
- Ramadhan, A. I., & Setiawan, E. B. 2023. Aspect Based Sentiment Analysis on Social Media Using Convolutional Neural Network (CNN) Method. *Building of Informatics, Technology and Science (BITS)*, 4(4), 1828–1836. <https://doi.org/10.47065/bits.v4i4.3103>.
- Ramdhani, M. A., Maylawati, D. S., & Mantoro, T. 2020. Indonesian News Classification using Convolutional Neural Network. *Indonesian Journal of Electrical Engineering and Computer Science*, 19(2), 1000–1009. <https://doi.org/10.11591/ijeecs.v19.i2.pp1000-1009>.
- Ranti, K. S., & Girsang, A. S. 2020. Indonesian Sarcasm Detection using Convolutional Neural Network. *IJETER* 8(9). <https://doi.org/10.30534/ijeter/2020/10892020>
- Sari, U. C., & Ulfiana, D. 2021. Pelatihan Online Analisis Laju Erosi Menggunakan Aplikasi QGIS Bagi Mahasiswa. *Jurnal Pengabdian Vokasi*, 2(1), 61-65.
- Shumaly, S., Yazdinejad, M., & Guo, Y. 2021. Persian sentiment analysis of an online store independent of pre-processing using convolutional neural network with fastText embeddings. *PeerJ. Computer science*, 7, e422. <https://doi.org/10.7717/peerj-cs.422>
- Umer, M., Imtiaz, Z., Ahmad, M., Nappi, M., Medaglia, C., Choi, G. S., & Mehmood, A. 2023. Impact of convolutional neural network and FastText embedding on text classification. *Multimedia Tools and Applications*, 82(4), 5569-5585.
- Yojana, Y. 2023. Tingkat Penerimaan Learning Management System (LMS) Pada Peserta Pelatihan Dasar Pegawai Negeri Sipil Kementerian Kesehatan. *Diklat Review: Jurnal manajemen pendidikan dan pelatihan*, 7(1), 138-148.