

BAB II

TINJAUAN PUSTAKA

Dalam bab tinjauan pustaka ini berisi tentang penjelasan mengenai beberapa penelitian tentang pustaka yang digunakan untuk membahas tentang Sistem Prediksi Jumlah Penumpang di Bandar Udara Juanda Surabaya dengan Metode *Double Exponential Smoothing* sebagai berikut.

2.1 Penelitian Terdahulu

Penelitian terdahulu adalah upaya peneliti untuk mencari perbandingan dan selanjutnya untuk menemukan inspirasi baru untuk penelitian selanjutnya di samping itu kajian terdahulu membantu penelitian dalam memposisikan penelitian serta menunjukkan orisinalitas dari penelitian. Secara umum pengertian penelitian terdahulu adalah sumber lampau dari hasil penelitian yang nantinya diusahakan oleh peneliti untuk membandingkan penelitian yang akan dilaksanakan. Penelitian terdahulu juga bisa berfungsi sebagai sumber inspirasi yang nantinya membantu pelaksanaan penelitian. Selain itu peneliti juga bisa memeriksa apa yang kurang dan kelebihan untuk dikembangkan. Sehingga ilmuwan juga bisa membuat sebuah penelitian yang orisinal/baru karena tahu mana yang sudah ditemukan dan mana yang belum. Pada bagian ini peneliti mencantumkan berbagai hasil penelitian terdahulu yang terkait dengan penelitian yang hendak dilakukan, kemudian membuat ringkasannya, baik penelitian yang sudah terpublikasikan atau belum terpublikasikan (skripsi, tesis, disertasi dan sebagainya). Dengan melakukan langkah ini, maka akan dapat dilihat sejauh mana orisinalitas dan posisi penelitian yang hendak dilakukan. Tabel 2.1 adalah tabel state of art dari penelitian sebelumnya.

Tabel 2.1 State of Art

Judul Penelitian dan Penulis	Metodologi Penelitian	Hasil
Sistem Prediksi Calon Penumpang Pesawat pada Bandar Udara Internasional Juanda Menggunakan Metode Double Exponential Smoothing (Lestari, 2015)	Dataset website Badan Pusat Statistik, yaitu bps.go.id.	Metode Double Exponential Smoothing menghasilkan prediksi dengan akurasi yang tinggi hingga 96.855%.
Prediksi Jumlah Penumpang di Bandar Udara Juanda Menggunakan Metode Support Vector Regression dengan Particle Swarm Optimization (Hidayat, 2018)	Data diambil dari bulan Januari 2010 hingga bulan Desember 2015.	MAPE terbaik 6.1210% metode Moving Average. Disusul MAPE sebesar 6.2559% menggunakan metode SVR-PSO.
Prediksi Lalu-Lintas Penumpang Bandar Udara Soekarno-Hatta dengan Teknik Time-Series Trend Forecasting (pahala,2019)	Data penumpang Bandar Udara Soekarno Hatta	MAPE antara 1.89 hingga 9.75 persen.

Tabel lanjutanl 2.1

Model Peramalan Hibrida untuk Prediksi Jumlah Penumpang Udara dan Volume Kargo di Indonesia (Sulistyowati, 2018)	Data sekunder dari (BPS).	Pemodelan menggunakan NN (ARIMAX-NN dan TSR-NN) data testing menghasilkan peramalan lebih baik dibandingkan model hibrida menggunakan SVR (TSR-SVR dan ARIMAX-SVR).
--	---------------------------	---

Sumber : <https://scholar.google.com>

2.2 Prediksi

Pengertian prediksi dengan cara menyamakan atau membentuk model sistem yang telah dipelajari dari data sebelumnya. Kemudian dipelajari atau memprediksi yang terjadi sebelumnya dengan mengadakan eksperimen secara teliti dan juga terstruktur dengan cara numerik memakai komputer. Dalam melakukan suatu prediksi yang sempurna diperlukan diantaranya: model dari masalah yang dipelajari dari sistem yang ada dalam komputer yang telah dijadikannya alat pengukur dalam melakukan sebuah eksperimen. Di dalam persoalan permodelan diperlukannya kemampuan konseptual informatika dan teori peluang, sedangkan di dalam hal-hal menggunakan komputer diperlukannya metode numerik atau pengetahuan komputasi lainnya, sehingga dapat dihasilkannya algoritma ataupun program-program komputer yang efisien.

Tahapan dalam melakukan prediksi adalah: mengumpulkan dataset beberapa tahun sebelumnya, merancang dan mendesain sistem termasuk variabel dependen

dan variabel independen, membuat perhitungan secara konsep untuk prediksi, dan menerapkan algoritma atau model untuk memprediksi data.

2.3 Double Exponential Smoothing

Double Exponential Smoothing ialah metode model-model linier yang telah ditemukan oleh Brown sebagai alat mengatasi sebuah perbedaan di antara data aktual dengan nilai-nilai prediksi. Metode pemulusan ini mempunyai dua nilai dari data-data yang ada unsur trend, perbedaan dari nilai-nilai pemulus tunggal dan ganda dijumlahkan ke nilai pemulsaan dan disesuaikannya untuk trend. Pada metode Double Exponential Smoothing melakukan progres smoothing sebanyak dua kali.

1. Rumus menentukan exponential tunggal (S'_t)

$$S'_t = \alpha X_t + (1 - \alpha)S'_{t-1} \dots\dots\dots (2.1)$$

2. Rumus menentukan exponential ganda (S''_t)

$$S''_t = \alpha S'_t + (1 - \alpha)S''_{t-1} \dots\dots\dots (2.2)$$

3. Rumus menentukan kontanta atau trend (α_t)

$$\alpha_t = S'_t + (S'_t - S''_t) = 2 S'_t - S''_t \dots\dots\dots (2.3)$$

4. Rumus menentukan slope (b_t)

$$b_t = \frac{\alpha}{1-\alpha} (S'_t - S''_t) \dots\dots\dots (2.4)$$

5. Rumus menentukan Besaran prediksi

$$S_{t+m} = \alpha_t + b_t (m) \dots\dots\dots (2.5)$$

Keterangan:

δ_{t+m} = Nilai ramalan m periode ke depan

m = Jarak-jarak yang akan di ramalkan

X_t = Nilai-nilai actual priode ke-t

δ^1_t = Nilai-nilai eksponential smothing periode ke - t

δ^{11}_t = Nilai-nilai double eksponential smothing periode ke - t

α = parameter exponential smothing yang besarnya $0 < \alpha$

Metode Double Smoothing Eksponential dapat diterapkan untuk sistem prediksi calon penumpang sehingga pihak dapat melakukan evaluasi pada setiap penerbangan dan dapat melakukan penjadwalan.

Dengan menggunakan sistem ini, monitoring lalu lintas darat dan manajemen aktivitas lebih terkontrol. Dan dapat memaksimalkan keberangkatan pesawat dan meminimalisir kemacetan lalu lintas darat. Sistem ini dapat melakukan komunikasi dengan pihak penerbangan untuk manajemen pesawat sehingga pesawat yang kekurangan penumpang tidak mengalami kerugian operasional. Selain itu penumpang dapat memilih pesawat yang tepat sesuai jadwal sehingga meminimalisir delay atau pergantian pesawat.

2.4 PHP

PHP atau (*Hypertext Preprocessor*) merupakan bahasa pemrograman script yang telah ditanam dalam sisi-sisi servers. Prosesor PHP dijalankan di Windows maupun Linux. Ketika halaman terbuka dan tercantum kodenya, prosesor tersebut akan mengartikan dan eksekusi perintahnya di halaman tersebut, dan selanjutnya memperlihatkan hasil ke dalam browsernya untuk halaman HTML normal. Karena penerjemah tersebut terjadi dalam server, halamannya ditulis menggunakan PHP dapat kita lihat dengan segala jenis browser, di sistem operasi apapun.

2.5 Entity Relationship Diagram (ERD)

ERD (Entity Relationship Diagram) adalah suatu model untuk menjelaskan hubungan antar data dalam basis data berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi. ERD untuk memodelkan struktur data dan hubungan antar data, untuk menggambarannya digunakan beberapa notasi dan simbol. Pada dasarnya ada tiga komponen yang digunakan, yaitu :

2.5.1 Entitas

Entiti merupakan objek yang mewakili sesuatu yang nyata dan dapat dibedakan dari sesuatu yang lain. Simbol dari entiti ini biasanya digambarkan dengan persegi panjang.

2.5.2 Atribut

Setiap entitas pasti mempunyai elemen yang disebut atribut yang berfungsi untuk mendeskripsikan karakteristik dari entitas tersebut. Isi dari atribut mempunyai sesuatu yang dapat mengidentifikasi isi elemen satu dengan yang lain. Gambar atribut diwakili oleh simbol elips.

a. Atribut Key

Atribut Key adalah satu atau gabungan dari beberapa atribut yang dapat membedakan semua baris data (Row/Record) dalam tabel secara unik. Dikatakan unik jika pada atribut yang dijadikan key tidak boleh ada baris data dengan nilai yang sama. Contoh : Nomor pokok mahasiswa (NPM), NIM dan nomor pokok lainnya

b. Atribut simple

Atribut simple adalah atribut yang bernilai atomic, tidak dapat dipecah/dipilah lagi. Contoh : Alamat, penerbit, tahun terbit, judul buku.

c. Atribut Multivalued

Atribut multivalued adalah nilai dari suatu attribute yang mempunyai lebih dari satu (multivalued) nilai dari attribute yang bersangkutan. Contoh : dari sebuah buku, yaitu terdapat beberapa pengarang.

d. Atribut Composite

Atribut composite adalah suatu atribut yang terdiri dari beberapa atribut yang lebih kecil yang mempunyai arti tertentu yang masih bisah dipecah lagi atau mempunyai sub attribute. Contoh : dari entitas nama yaitu nama depan, nama tengah, dan nama belakang

e. Atribut Derivatif

Atribut derivatif adalah atribut yang tidak harus disimpan dalam database Ex. Total. atau atribut yang dihasilkan dari atribut lain atau dari suatu relationship. Atribut ini dilambangkan dengan bentuk oval yang bergaris putus-putus

2.5.3 Hubungan / Relasi

Hubungan antara sejumlah entitas yang berasal dari himpunan entitas yang berbeda.

2.5.4 Derajat relasi atau kardinalitas rasio

Menjelaskan jumlah maksimum hubungan antara satu entitas dengan entitas lainnya

2.5.5 One to One (1:1)

Setiap anggota entitas A hanya boleh berhubungan dengan satu anggota entitas B, begitu pula sebaliknya.

2.5.6 One to many (1:M / Many)


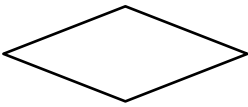
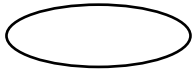

Setiap anggota entitas A dapat berhubungan dengan lebih dari satu anggota entitas B tetapi tidak sebaliknya.

2.5.7 Many to Many (M:M)

Setiap entitas A dapat berhubungan dengan banyak entitas himpunan entitas B dan demikian pula sebaliknya

Simbol - simbol yang digunakan dalam DFD antara lain:

Tabel 2.2 Tabel Simbol-simbol ERD

Simbol	Keterangan
	<p>Entitas Suatu objek yang dapat diidentifikasi dalam lingkungan pemakai</p>
	<p>Relasi Menunjukkan adanya hubungan di antara sejumlah entitas yang berbeda.</p>
	<p>Atribut Mendeskripsikan karakter entitas.</p>
	<p>Garis Penghubung antara relasi dengan entitas, relasi dan entitas dengan atribut.</p>

2.6 Flowchart

Flowchart adalah adalah suatu bagan dengan simbol-simbol tertentu yang menggambarkan urutan proses secara mendetail dan hubungan antara suatu proses (instruksi) dengan proses lainnya dalam suatu program.

Dalam perancangan flowchart sebenarnya tidak ada rumus atau patokan yang bersifat mutlak (pasti). Hal ini didasari oleh flowchart (bagan alir) adalah sebuah gambaran dari hasil pemikiran dalam menganalisa suatu permasalahan dalam komputer. Karena setiap analisa akan menghasilkan hasil yang bervariasi antara satu dan lainnya. Kendati begitu secara garis besar setiap perancangan flowchart selalu terdiri dari tiga bagian, yaitu input, proses dan output. Simbol - simbol yang dipakai dalam *flowchart* dibagi 3 kelompok:

- a. *Flowdirectionsymbols*, digunakan untuk menghubungkan simbol satu dengan yang lain dan disebut juga *connecting line*.
- b. *Processingsymbols*, Menunjukkan jenis operasi pengolahan dalam suatu proses / prosedur.
- c. *Input / Outputsymbol*, Menunjukkan jenis peralatan yang digunakan sebagai media *input* atau *output*.

Berikut Tabel 2.3 adalah beberapa simbol yang digunakan dalam menggambar suatu *flowchart*:

2.7 Data Flow Diagram (DFD)

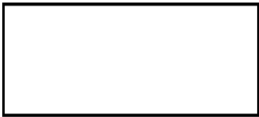
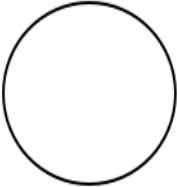
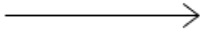
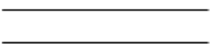
DFD merupakan rancangan model Logika (*logic model*) dari sistem informasi lebih menjelaskan kepada *user* bagaimana nantinya fungsi - fungsi di sistem informasi secara logika akan bekerja. Secara umum DFD (Data Flow Diagram) adalah suatu langkah atau metode untuk membuat sebuah perancangan sistem yang mana berorientasi pada alur data yang bergerak kesebuah sistem lainnya. Dalam membuat Sistem Informasi ini, DFD sering dipakai.

DFD dibuat oleh para analis untuk membuat sebuah sistem dengan baik. Dimana DFD ini nantinya dikasihkan kepada para programmer untuk memulai

proses coding. Yang mana para programmer ini melakukan sebuah coding sesuai dengan DFD yang dibuat oleh para analis sebelumnya. Model logika dapat digambarkan menggunakan diagram arus data (*Data Flow Diagram*).

DFD menggambarkan arus data dari suatu sistem, baik sistem lama maupun sistem baru secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut berada (Kusrini, 2007).

Tabel 2.4 Tabel Simbol-simbol DFD

Simbol	Keterangan
	Entitas (<i>Ecternal Entity</i>) Simbol terminator menggambarkan asal atau tujuan data di luar sistem
	Proses Simbol lingkaran menunjukkan transformasi dari masukan menjadi keluaran. Umumnya didefinisikan dengan kata tunggal atau kalimat sederhana.
	Arus Data (<i>Data Flow</i>) Menggambarkan aliran data yaitu gerakan paket data atau informasi dari suatu bagian ke bagian lain dari sistem. Ujung panah meunjukkan arah data bergerak.
	<i>Data Store</i> Menggambarkan tempat data disimpan.