BAB II
TINJAUAN PUSTAKA DAN DASAR TEORI

2.1. Tinjauan Pustaka

Dalam rentang waktu satu dekade, terdapat sejumlah studi terkait model ujian berbasis kertas dibandingkan dengan ujian berbasis komputer. Sebuah penelitian tentang tes bahasa Inggris mendapatkan hasil yang komparatif, namun terdapat bias gender. Pada penelitian tersebut peserta tes laki-laki cenderung memilih ujian berbasis komputer, sementara peserta tes berjenis kelamin perempuan lebih memilih ujian berbasis kertas (Coniam, 2006).

Pada penelitian lain, diperoleh hasil tes setara untuk kedua jenis model berbasis kertas dan berbasis komputer. Namun, hasil ujian berbasis komputer bersifat lebih stabil dan konsisten untuk beberapa kali pelaksanaan ujian dengan peserta yang sama. Penelitian ini lebih condong pada dukungan untuk model ujian berbasis komputer (Piaw, 2012).

Dalam penelitian lain, dilakukan pengukuran penggunaan kertas coretan untuk kedua model ujian. Ternyata hasil penelitian tersebut mengungkap bahwa peserta ujian berbasis kertas lebih banyak menghabiskan kertas coretan apabila dibandingkan dengan peserta tes berbasis komputer. Dengan demikian dapat disimpulkan bahwa ujian berbasis komputer lebih menghemat penggunaan sumber daya alam (Prisacari dan Danielson, 2017).

Penelitian model ujian terbaru mendapatkan hasil yang setara untuk kedua jenis ujian. Dalam penelitian tersebut dilakukan sebuah tes menulis dalam bahasa Inggris. Mayoritas responden lebih memilih ujian berbasis komputer. Hal ini disebabkan oleh kondisi responden yang sudah terbiasa dengan penggunaan perangkat teknologi informasi (Chan dkk., 2018).

Sebuah penelitian tentang *data mining* dikaitkan dengan domain pendidikan dilakukan dengan cara meninjau sejumlah artikel jurnal dalam konteks *educational data mining* (EDM). Berdasarkan analisis, diperoleh informasi bahwa penelitian dalam EDM dikembangkan menjadi beberapa tema, antara lain studi

berorientasi pada interaksi pembelajaran, evaluasi pembelajaran, dan rekomendasi dan pemulihan media pendidikan. Penelitian tersebut menyajikan perspektif, indentifikasi tren, dan arah penelitian potensial, misalnya perilaku, kolaborasi, interaksi dan kinerja dalam pengembangan proses pembelajaran (Rodrigues, 2018).

Adapun penelitian lain tentang *data mining* telah dilakukan pada sebuah universitas untuk memprediksi tingkat kesuksesan studi mahasiswa. Algoritma *data mining* sangat cocok untuk data berukuran besar, sedangkan data mahasiswa terkait perkuliahan dikategorikan berukuran kecil. Penelitian tersebut menitikberatkan pada *data mining* untuk kumpulan data kecil. Untuk menjawab pertanyaan penelitian, dilakukan pembandingan dua buah piranti *data mining*, yaitu Excel dan WEKA. Kesimpulan yang diperoleh sangat menjanjikan dan mendorong perguruan tinggi untuk melibatkan piranti *data mining* sebagai bagian penting dalam sistem manajemen pengetahuan pendidikan tinggi (Natek dan Zwilling, 2014).

2.2. Dasar Teori

2.2.1. Teori Pengembangan Sistem Informasi

Untuk mendukung penelitian yang dilakukan, diperlukan teori tentang pengembangan sistem informasi. Pengembangan sistem aplikasi untuk pra-pemrosesan data mengacu pada *System Development Life Cycle* (SDLC) model *Waterfall* yang terdiri atas enam tahap yaitu *requirement*, *analysis*, *design*, *coding*, *testing*, dan *operation* (Royce, 1970). Dalam tahap *requirement*, dilakukan penyelidikan tentang kebutuhan sistem sehingga menghasilkan *Software Requirements Specification* (SRS). Dalam tahap analisis, dilakukan pemodelan fungsi sistem dan pemodelan data. Model fungsi sistem secara hirarki dituangkan dalam bentuk *Data Flow Diagram* (DFD). DFD level 0 disebut juga *Data Context Diagram* (DCD). Sedangkan model data digambarkan dalam bentuk *Entity Relationship Diagram* (ERD).

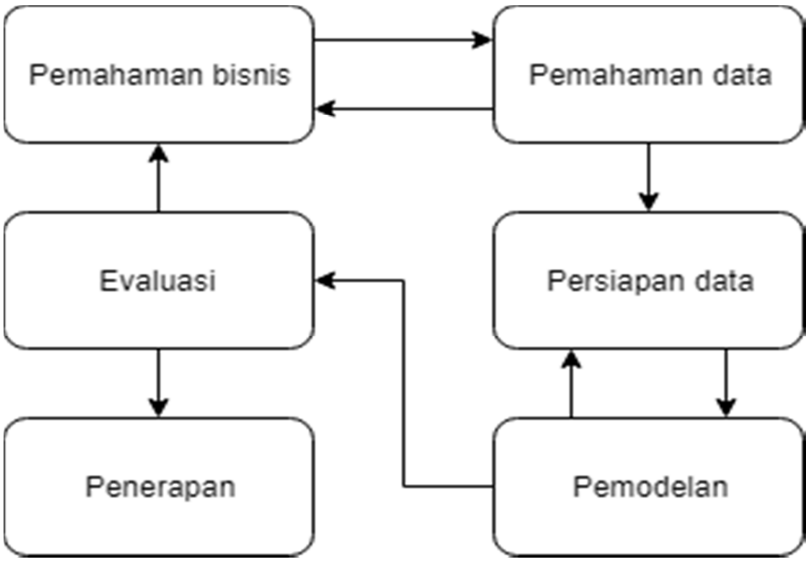
Dalam tahap desain, dilakukan perancangan algoritma berdasarkan DFD dan perancangan struktur data berdasarkan ERD. Selanjutkan dalam tahap *coding*, dilakukan implementasi rancangan ke dalam pemrograman komputer. Tahap *testing* merupakan tahap pengujian terhadap perangkat lunak yang telah dibuat. Secara sederhana terdapat model pengujian *white box* dan *black box*. Dalam model *white box* dilakukan pengujian atas rincian setiap algoritma program di dalam sistem. Sedangkan dalam model *black box* dilakukan pengujian atas setiap fungsi di dalam sistem dalam melakukan tugasnya sesuai spesifikasi (Pressman, 2015). Adapun tahap akhir *Waterfall* yaitu *operation* merupakan tahap saat perangkat lunak tersebut dijalankan dengan kondisi data dan lingkungan nyata.

2.2.2. Teori *Data Mining*

Data mining merupakan upaya untuk mendapatkan pengetahuan tersembunyi dari data masif melalui berbagai macam algoritma. *Data mining* mengkolaborasikan berbagai disiplin antara lain sistem basis data, statistika, pembelajaran mesin, dan pengenalan pola. *Data mining* sendiri merupakan bagian dari proses penemuan pengetahuan yang lebih besar, yang mencakup tugas-tugas pra-pemrosesan dan pasca-pemrosesan. Tugas pra-pemrosesan misalnya ekstraksi data, pembersihan data, penggabungan data, reduksi data, dan konstruksi fitur. Tugas pasca-pemrosesan misalnya interpretasi pola dan model, pembangkitan dan konfirmasi asumsi. *Knowledge discovery* dan *data mining* merupakan dua proses yang iteratif dan interaktif (Zaki dan Meira, 2014).

Proses dalam *data mining* terdiri atas 6 tahap. Pada tahap pertama, pemahaman bisnis dilakukan untuk menentukan kesesuaian penerapan *data mining* dalam suatu persoalan. Kedua, pemahaman data dilakukan untuk menentukan kecocokan data dalam proses lanjutan. Kedua tahap tersebut dapat dilakukan bersamaan dan saling melengkapi. Ketiga, persiapan data dilakukan dengan cara mentransformasi data sedemikian rupa agar algoritma pembelajaran mesin dapat memproduksi model. Keempat, pemodelan dilakukan untuk membangun model data. Tahap persiapan data dan pemodelan dilakukan secara

iteratif dan interaktif. Kelima, evaluasi dilakukan untuk memastikan model data sesuai dengan kondisi nyata. Jika model tidak cocok, maka proses pemahaman bisnis diulang. Jika model cocok, maka dilanjutkan ke tahap penerapan (Witten, 2017). Siklus *data mining* ditunjukkan pada Gambar 2.1.



Gambar 2.1 Siklus *Data mining*

Teknik *data mining* dapat digunakan dalam analisis data. Teknik *data mining* dapat berupa klasifikasi, asosiasi, ataupun *clustering*. Teknik *data mining* klasifikasi cocok untuk memetakan suatu instans informasi ke dalam kategori kelas yang telah didefinisikan berdasarkan kombinasi nilai tertentu dari atribut instans tersebut (Han dkk., 2011).

Teknik klasifikasi sederhana dapat dilakukan dengan pembelajaran pohon keputusan. Pembelajaran pohon keputusan merupakan sebuah metode untuk melakukan pendekatan terhadap fungsi target yang bernilai diskrit. Fungsi yang dipelajari tersebut direpresentasikan dengan pohon keputusan. Pohon yang dipelajari juga dapat direpresentasikan sebagai kumpulan aturan *if-then* untuk meningkatkan keterbacaan manusia. Pohon keputusan mengklasifikasikan contoh dengan menyortir pohon dari simpul akar ke beberapa simpul daun, yang

menyediakan klasifikasi instans. Pada setiap simpul dalam pohon dilakukan pengujian beberapa atribut dari instans, dan setiap cabang yang turun dari simpul tersebut sesuai dengan salah satu nilai yang mungkin untuk atribut tersebut. Suatu instans diklasifikasikan dengan cara mulai pada simpul akar pohon dilakukan pemeriksaan nilai atribut yang ditentukan oleh simpul ini. Kemudian fokus berpindah ke cabang pohon yang sesuai dengan nilai atribut dalam contoh yang diberikan. Proses ini kemudian diulang untuk subpohon yang berakar pada simpul baru (Mitchell, 1997).

Karakteristik persoalan yang cocok untuk diselesaikan dengan pembelajaran pohon keputusan terdiri atas lima buah syarat. Pertama, instans direpresentasikan sebagai pasangan atribut-nilai, contohnya atribut temperatur memiliki kemungkinan nilai yang saling lepas (panas, sedang, dingin). Kedua, fungsi target memiliki nilai keluaran diskrit, misalnya atribut keputusan memiliki kemungkinan nilai yang saling lepas (ya, tidak). Ketiga, memungkinkan deskripsi ekspresi yang saling lepas. Keempat, data latih mungkin memiliki *error*, baik *error* dalam pengklasifikasian sampel latih, maupun *error* dalam nilai atribut yang mendeskripsikan sampel tersebut. Kelima, data latih mungkin berisi nilai atribut yang hilang (Mitchell, 1997).

2.2.3. Algoritma *Decision Tree* C4.5

Decision Tree C4.5 merupakan salah satu algoritma yang dapat digunakan dalam klasifikasi. Dalam C4.5, dilakukan pemangkasan simpul agar terbentuk model pohon yang lebih efisien. Pertama, bangun pohon keputusan dari *training set*, kembangkan simpul sedemikian hingga data latih menjadi fit dan terdefinisi, dan memungkinkan terjadinya *overfitting*. *Overfitting* adalah kondisi di mana klasifikasi terlalu kaku dan klasifikasi dapat gagal apabila ada nilai yang hilang pada instans yang tidak dapat memenuhi alur pohon atau aturan. Kedua, lakukan konversi pohon yang dipelajari menjadi himpunan aturan yang ekivalen dengan cara merumuskan sebuah aturan untuk setiap alur dari simpul akar ke simpul daun. Ketiga, pangkas setiap aturan dengan cara menghapus prekondisi yang dapat memperbaiki perkiraan akurasi. Keempat, urutkan aturan-aturan yang

dipangkas berdasarkan perkiraan akurasi ketika mengklasifikasikan instans yang berurutan (Quinlan, 1993).

Dalam pembentukan pohon keputusan diperlukan pengukuran kuantitatif yang tepat atas seberapa berarti sebuah atribut yang berasosiasi dengan simpul. Sebuah properti bernama *information gain* digunakan untuk mengukur seberapa penting sebuah atribut memisahkan sampel-sampel latih sesuai dengan klasifikasi target. Untuk mendefinisikan *information gain* secara tepat, digunakan istilah dalam teori informasi yaitu entropi. Entropi mencerminkan seberapa murni koleksi sampel yang berubah-ubah (Shannon, 1951). Nilai entropi minimum 0 mencerminkan informasi sebuah nilai atribut murni, artinya setiap instans pasti dipetakan langsung ke sebuah kelas. Nilai entropi maksimum 1 mencerminkan meratanya sebaran setiap nilai dalam sebuah atribut, artinya jumlah instans untuk setiap nilai adalah sama, sehingga tingkat kepastian pemetaan kelas menjadi berkurang. Untuk menghitung entropi dapat digunakan Formula 2.1 di mana S adalah koleksi sampel, c adalah banyaknya partisi sampel, dan p_i adalah proporsi ke-i dari sampel (Mitchell, 1997).

$$Entropy(S) \equiv \sum_{i=1}^{c} p_i \log_2 p_i \tag{2.1}$$

Rumus untuk menghitung *information gain* ditunjukkan pada Formula 2.2 di mana *gain(S,A)* adalah *information gain* atribut A relatif terhadap sampel S. *Values* adalah himpunan semua kemungkinan nilai dari atribut A. S_v sendiri merupakan himpunan bagian dari S yang mana atribut A memiliki nilai v (Mitchell, 1997).

$$Gain(S, A) \equiv Entropy(S) \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v) \tag{2.2}$$

Split information digunakan untuk menghitung atribut penentu yang sangat bergantung pada seberapa luas dan tingkat keseragaman atribut dalam membagi kelompok data. Untuk menghitung *split information* digunakan Formula 2.3 di mana S_1 sampai S_c adalah c buah himpunan bagian dari sampel yang dihasilkan dari proses partisi S menjadi sebanyak c buah nilai atribut A. *Split information* sebenarnya adalah entropi S untuk atribut A (Mitchell, 1997).

$$SplitInformation(S,A) \equiv \sum_{i=1}^{c} \frac{|S_i|}{|S|} \log_2 \frac{|S_i|}{|S|} \tag{2.3}$$

Pada algoritma C4.5, penentuan atribut keputusan diukur dengan *gain ratio*. *Gain ratio* dihitung sebagai rasio *information gain* dengan *split information* sebagaimana tertulis dalam Formula 2.4 (Mitchell, 1997). Namun, ada isu praktis yang muncul yaitu dalam seleksi atribut yang pembaginya bernilai nol atau sangat kecil. Hal ini membuat *gain ratio* menjadi tidak terdefinisi atau terlalu besar untuk atribut yang memiliki nilai sama pada hampir semua anggota S (Quinlan, 1993).

$$GainRatio(S,A) \equiv \frac{Gain(S,A)}{SplitInformation(S,A)} \tag{2.4}$$

Proses pemilihan atribut untuk simpul dilakukan berulang sehingga tercapai salah satu dari dua buah kondisi. Kondisi pertama, setiap atribut telah dimasukkan dalam model pohon. Kondisi kedua, sampel latih yang berasosiasi dengan simpul akhir memiliki nilai atribut target yang sama atau dengan kata lain, entropi atribut-atribut tersebut bernilai nol (Mitchell, 1997).

2.2.4. Teori Pengujian Data

Untuk mendukung proses analisis, diperlukan pengetahuan tentang pengujian data. Terdapat konvergensi antara teori dalam Statistika dan

Informatika dalam era *data science* (Ceri, 2018). Ahli statistika lebih fokus pada data yang diproses, sedangkan ahli Informatika lebih fokus pada proses yang dilakukan terhadap data. Terdapat dua kultur dalam penggunaan model statistika. Model data stokastik menggunakan data asli sedangkan model data algoritmik lebih mudah diolah dengan komputer (Breiman, 2001).

Dalam evaluasi disusun dua buah asumsi yang saling bertentangan tentang kondisi data, yang disebut asumsi pertama dan asumsi kedua. Setelah dilakukan pengujian, dapat diputuskan untuk menerima asumsi pertama dan menolak asumsi kedua ataukah diputuskan sebaliknya. Dalam evaluasi dan juga klasifikasi perlu diperhatikan dua jenis *error*, yaitu *false positive* dan *false negative*. *False positive* terjadi jika asumsi yang diterima sebenarnya salah, namun dianggap memiliki efek signifikan ketika asumsi tersebut dianggap aktual karena ada perubahan. Sedangkan *false negative* terjadi jika asumsi ditolak padahal sebenarnya benar (Downey, 2011). Akurasi klasifikasi dihitung dari jumlah *true positive* dan *true negative* dibagi total sampel uji (Han dkk., 2011).

Terkadang muncul persoalan di mana asumsi pertama benar namun terdapat sedikit perbedaan di antara dua kelompok data. Untuk mengatasi hal itu dilakukan *Cross Validation* yang menggunakan sebuah dataset untuk menghitung perbedaan dan menggunakan dataset lainnya untuk mengevaluasi asumsi pertama. Dataset pertama disebut kelompok data latih, sedangkan dataset kedua disebut kelompok data uji. *Cross Validation* 10 *folds* berarti sampel dibagi menjadi 10 partisi. Setiap partisi digunakan sebagai data uji untuk data latih yang merupakan gabungan dari 9 partisi lain. Pengujian dilakukan pada semua partisi dan diambil nilai rataan akurasi (Witten, 2017).

Pengujian juga dapat menggunakan *Percentage Split*. Pada pengujian *Percentage Split* dilakukan evaluasi untuk sebagian data saja. Misalnya, *Percentage Split* 66% berarti ada sebanyak 66% ukuran sampel digunakan sebagai data latih sedangkan sisanya 34% ukuran sampel digunakan untuk data uji (Witten, 2017).