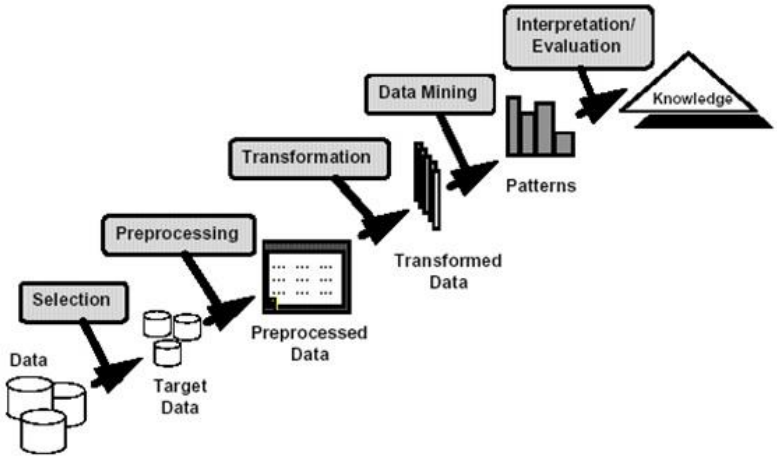
BAB II

LANDASAN TEORI

2.1. Data Mining

Tidak mengherankan bahwa *data mining*, sebagai subjek yang benar-benar interdisipliner, dapat didefinisikan dengan berbagai cara. Bahkan istilah penambangan data (*data mining*) tidak benar-benar menyajikan semua komponen utama untuk gambar. Untuk mempertahankan keterangan dari emas dan, kita akan menambang emas alih-alih untuk menyapu atau menambang tepat disebut "penambangan pengetahuan dari data," yang sayangnya agak panjang. Namun, istilah yang lebih pendek, penambangan pengetahuan mungkin tidak mencerminkan penekanan pada penambangan dari sejumlah besar data. Namun demikian, *mining* adalah istilah yang sangat jelas yang menggambarkan proses yang menemukan sejumlah kecil cuplikan berharga dari banyak bahan baku. Dengan demikian, kesalahan nama seperti membawa "*data*" dan "*mining*" menjadi pilihan populer. Selain itu, banyak istilah lain yang memiliki arti yang mirip dengan penambangan data misalnya, penambangan pengetahuan dari data, ekstraksi pengetahuan, analisis pola atau data, arkeologi data, dan pengerukan data. Banyak orang memperlakukan penambangan data sebagai sinonim untuk istilah lain yang populer digunakan, *knowledge discovery from data* (KDD) sementara yang lain melihat penambangan data hanya sebagai langkah penting dalam proses penemuan pengetahuan. Han menjabarkan proses *data mining* sebagai pada Gambar 2.1 (Han, 2012) berikut:



Gambar 2.1 Proses *data mining*

1. *Data cleaning* (untuk menghilangkan *noise* dan data yang tidak konsisten)
2. *Data integration* (di mana banyak sumber data dapat digabungkan)
3. Pemilihan data (di mana data yang relevan dengan tugas analisis diambil dari *database*)
4. Transformasi data (di mana data ditransformasikan dan dikonsolidasikan ke dalam bentuk yang sesuai untuk penambangan dengan melakukan operasi ringkasan atau agregasi)
5. *Data Mining* (proses penting di mana metode cerdas diterapkan untuk mengekstraksi pola data)
6. Evaluasi Pola (untuk mengidentifikasi pola yang benar-benar menarik yang mewakili pengetahuan berdasarkan langkah-langkah ketertarikan)
7. Presentasi pengetahuan (di mana teknik visualisasi dan representasi pengetahuan digunakan untuk menyajikan pengetahuan yang ditambang kepada pengguna)

Data mining adalah proses yang menggunakan teknik statistik, matematika, kecerdasan buatan, dan *machine learning* untuk mengekstraksi dan mengidentifikasi informasi yang bermanfaat dan pengetahuan yang terkait dari berbagai *database* besar (Turban, 2005). Dalam *data mining* terdapat empat pendekatan metode pelatihan (Han, 2011:24) yaitu:

a. ***Supervised learning***, pada dasarnya adalah sinonim untuk klasifikasi. Pengawasan dalam pembelajaran berasal dari contoh-contoh berlabel dalam kumpulan data pelatihan. Misalnya, dalam masalah pengenalan kode pos, satu set gambar kode pos tulisan tangan dan terjemahannya sesuai dengan terjemahan mesin yang dibaca menggunakan contoh pelatihan, yang mengawasi pembelajaran model klasifikasi.
b. ***Unsupervised learning***, pada dasarnya adalah sinonim untuk pengelompokan. Proses pembelajaran tidak diawasi karena contoh *input* tidak diberi label kelas. Biasanya, kami dapat menggunakan pengelompokan untuk menemukan kelas dalam data. Sebagai contoh, sebuah metode pembelajaran yang tidak diawasi dapat mengambil, asinput, aset dari gambar-gambar hasil tulisan tangan. Mengadopsi bahwa menemukan 10 kelompok data. Kluster ini dapat sesuai dengan masing-masing 10 digit 0 hingga 9. Namun, karena data pelatihan tidak diberi label, model yang dipelajari tidak dapat memberi tahu kami makna semantik dari kluster yang ditemukan.
c. ***Semi-Supervised learning*** adalah kelas teknik pembelajaran mesin yang menggunakan contoh yang berlabel dan tidak berlabel saat mempelajari suatu model. Dalam satu pendekatan, contoh berlabel digunakan untuk

mempelajari model kelas dan contoh tidak berlabel digunakan untuk memperbaiki batas antar kelas. Untuk masalah dua kelas, kita dapat menganggap kumpulan contoh yang dimiliki oleh satu kelas sebagai contoh positif dan yang dimiliki oleh kelas lain sebagai contoh negatif. Jika kita tidak mempertimbangkan contoh-contoh yang tidak berlabel, garis putus-putus adalah batas keputusan yang memartisi terbaik contoh-contoh positif dari contoh-contoh negatif. Dengan menggunakan contoh yang tidak berlabel, kita dapat memperbaiki batas keputusan ke garis yang solid. Selain itu, kita dapat mendeteksi bahwa dua contoh positif di sudut kanan atas, meskipun diberi label, kemungkinan kebisingan atau *outlier*.

d. ***Active learning*** pendekatan pembelajaran mesin yang memungkinkan pengguna memainkan peran aktif dalam proses pembelajaran. Pendekatan pembelajaran aktif dapat meminta pengguna (misalnya Pakar domain) untuk memberi label contoh, yang mungkin berasal dari sekumpulan contoh yang tidak berlabel atau disintesis oleh program pembelajaran. Tujuannya adalah untuk mengoptimalkan kualitas model dengan secara aktif memperoleh pengetahuan dari pengguna manusia, diberikan batasan pada berapa banyak contoh yang dapat mereka tanyakan pada label.

e. ***Reinforcement learning*** adalah area pembelajaran mesin yang berkaitan dengan bagaimana agen perangkat lunak harus mengambil tindakan dalam lingkungan untuk memaksimalkan gagasan imbalan kumulatif. RL mengaplikasikan pembelajaran *trial and error* untuk mencapai target yang diharapkan. Dalam menghadapi masalah RL akan mempelajari tingkah

laku melalui pembelajaran *trial and error* untuk berinteraksi dengan lingkungan yang dinamis.

2.2. Klasifikasi Teks

Klasifikasi teks merupakan proses menemukan pola baru yang belum terungkap sebelumnya. Klasifikasi teks dilakukan dengan memproses dan menganalisis data dalam jumlah besar. Dalam prosesnya, klasifikasi teks melibatkan struktur yang mungkin terdapat pada teks dan mengekstrak informasi yang relevan pada teks. Dalam menganalisis sebagian atau keseluruhan teks yang tidak terstruktur, klasifikasi teks mencoba mengasosiasikan sebagian atau keseluruhan satu bagian teks dengan yang lainnya berdasarkan aturan-aturan tertentu (Miller, 2015). Tantangan dari klasifikasi teks adalah sifat data yang tidak terstruktur dan sulit untuk menangani, sehingga diperlukan proses *text mining*. Diharapkan melalui proses *text mining*, informasi yang ada dapat dikeluarkan secara jelas di dalam teks tersebut dan dapat dipergunakan dalam proses analisis menggunakan alat bantu komputer (Witten dkk, 2016).

2.2.1 Ketepatan Klasifikasi Model

Tahapan pra-proses ini dilakukan agar dalam klasifikasi dapat diproses dengan baik. Tahapan dalam pra-proses teks adalah sebagai berikut:

a. *Case Folding*, merupakan proses untuk mengubah semua karakter pada teks menjadi huruf kecil. Karakter yang diproses hanya huruf ‘a’ hingga ‘z’ dan selain karakter tersebut akan dihilangkan seperti tanda baca titik (.), koma (,), dan angka. (Weiss, 2016)

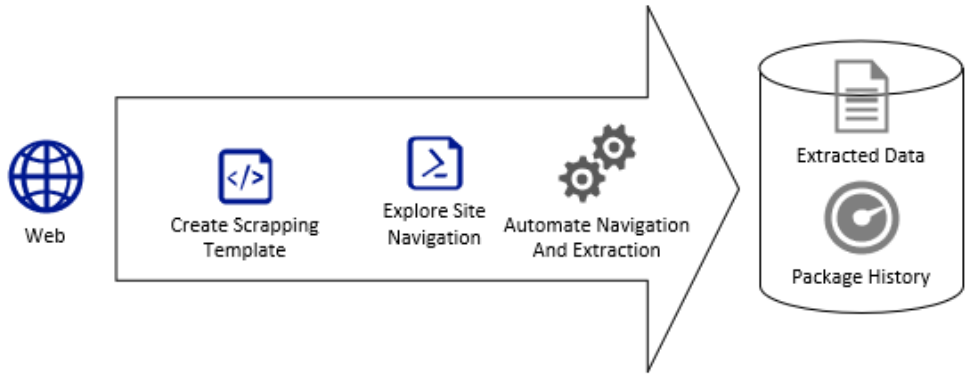
b. *Tokenizing*, merupakan proses memecah yang semula berupa kalimat menjadi kata-kata atau memutus urutan *string* menjadi potongan-potongan seperti kata-kata berdasarkan tiap kata yang menyusunnya. Sehingga dapat dikatakan mengembalikan kata penghubung.

c. *Stopwords*, yakni kosakata yang bukan merupakan kata unik atau ciri pada suatu dokumen atau tidak menyampaikan pesan apa pun secara signifikan pada kalimat (Dragut et al. 2016). Kosakata yang dimaksudkan tersebut adalah kata penghubung dan kata keterangan yang bukan merupakan kata unik misalnya "sebuah", "oleh", "pada", dan sebagainya.

d. *Stemming*, yakni proses untuk mendapatkan kata dasar dengan cara menghilangkan awalan, akhiran, sisipan, dan kombinasi dari awalan dan akhiran.

2.3. Web Scraping

Web Scraping (Turland, 2015) adalah proses pengambilan sebuah dokumen dari internet, umumnya berupa halaman-halaman web dalam bahasa *markup* seperti HTML atau XHTML, dan menganalisis dokumen tersebut untuk diambil data tertentu dari halaman tersebut yang digunakan untuk kepentingan lain. *Web scraping* memiliki sejumlah langkah, sebagai pada Gambar 2.2 (Turland, 2015) berikut:



Gambar 2.2 Langkah Web Scraping

a. *Create Scraping Template*: Pembuat program mempelajari dokumen HTML dari *website* yang akan diambil informasinya untuk *tag* HTML yang mengapit informasi yang akan diambil.

b. *Explore Site Navigation*: Pembuat program mempelajari teknik navigasi pada website yang akan diambil informasinya untuk ditirukan pada aplikasi *web scraper* yang akan dibuat.

c. *Automate Navigation and Extraction*: Berdasarkan informasi yang didapat pada langkah 1 dan 2 di atas, aplikasi *web scraper* dibuat untuk mengotomatisasi pengambilan informasi dari website yang ditentukan.

d. *Extracted Data and Package History*: Informasi yang didapat dari langkah 3 disimpan dalam tabel atau tabel-tabel *database*.

2.4. Naïve Bayes Classifier

Naïve Bayes merupakan metode klasifikasi yang berdasarkan pada Teorema Bayes. Dinamakan ‘*naïve’* karena algoritma mengasumsi bahwa fitur pengukurannya bersifat independen dengan satu sama lain, walaupun biasanya sifat ini tidak pernah benar. Teorema Bayes merupakan teorema yang mengacu konsep

probabilitas bersyarat (Turland, 2014). Dalam teori probabilitas, probabilitas bersyarat (*conditional probability*) adalah ukuran dari probabilitas suatu peristiwa yang terjadi mengingat bahwa peristiwa lain telah terjadi (dengan asumsi, anggapan, pernyataan atau bukti) terjadi.

Teorema Bayes dapat dinotasikan pada persamaan berikut:

$$P(A|B) = \frac{P(A)P(B|A)}{P(B)} \qquad ...(2.1)$$

Keterangan:

P(A|B):Probabilitas akhir bersyarat (*conditional probability*) suatu hipotesis A terjadi jika diberikan bukti (*evidence*) B terjadi.

P(B|A): Probabilitas sebuah bukti B terjadi jika hipotesis A terjadi.

P(A): Probabilitas awal (*priori*) hipotesis A terjadi tanpa memandang bukti apa pun.

P(B): Probabilitas awal (*priori*) bukti B terjadi tanpa memandang hipotesis/bukti yang lain.

Metode *Naïve bayes* yang sering disebut sebagai *Naïve bayes classification* (NBC), merupakan salah satu metode yang dapat mengklasifikasikan teks. Kelebihan NBC adalah sederhana tetapi memiliki akurasi yang tinggi. Dalam algoritma NBC setiap dokumen direpresentasikan dengan pasangan atribut "a_1, a_2, a_3,..., a_n" di mana a_1 adalah kata pertama, a_2 adalah kata kedua dan seterusnya. Sedangkan V adalah himpunan kategori berita. Pada saat klasifikasi algoritma akan mencari probabilitas tertinggi dari semua kategori dokumen yang diujikan (VMAP). Adapun persamaan VMAP adalah sebagai berikut:

$$V_{map} = \underset{v_j \in v}{argmax}\ P(v_j \mid a_1, a_2, \ldots\ldots, a_n) \qquad \text{...(2.2)}$$

Dengan menggunakan teorema Bayes, maka persamaan (2.2) dapat ditulis menjadi,

$$V_{map} = \underset{v_j \in v}{argmax} \frac{P(a_1, a_2, \ldots\ldots, a_n \mid v_j)\ P(v_j)}{P(a_1, a_2, \ldots\ldots, a_n)}$$

Karena nilai P(a_1, a_2, a_3,... a_n) untuk semua v_j besarnya sama maka nilainya dapat diabaikan, sehingga persamaan di atas menjadi:

$$V_{map} = \underset{v_j \in v}{argmax}\ P(a_1, a_2, \ldots\ldots, a_n \mid v_j)\ P(v_j)$$

Naïve bayes classifier menyederhanakan hal ini dengan mengasumsikan bahwa di dalam setiap kategori, setiap atribut bebas bersyarat satu sama lain (Turland, 2014). NBC didapatkan dari frekuensi kata-kata setiap berita yang diakumulasi berdasarkan kata terbanyak dari berita yang sudah ditentukan. Dengan kata lain persamaan di atas dapat dituliskan sebagai berikut:

$$\mathrm{P}(a_1, a_2, \ldots\ldots, a_n \mid v_j) = \prod_i P(a_1 \mid v_j)$$

Kemudian apabila persamaan di atas disubstitusikan ke persamaan V_{map} sebelumnya, maka akan menghasilkan persamaan:

$$V_{map} = \underset{v_j \in v}{argmax}\ P(v_j)\ \prod_i P(a_1 \mid v_j) \qquad \text{...(2.3)}$$

Keterangan:

V_{MAP}: semua kategori yang diujikan

V_j: Kategori berita, dengan:

$P(a_i|V_J)$: probabilitas a_i pada kategori V_j

$P(V_j)$: probabilitas dari V_j

Nilai $P(V_j)$ dihitung pada saat data *training*, didapat dengan rumus sebagai berikut:

$$P(v_j) = \frac{|\, doc\ j \,|}{|training|} \qquad ...(2.4)$$

Di mana |*doc j*| merupakan jumlah dokumen (artikel berita) yang memiliki kategori *j* dalam *training*. Sedangkan |*training*| merupakan jumlah dokumen (artikel berita) dalam contoh yang diguna-kan untuk *training*. Untuk probabilitas kata a_i untuk setiap kategori P(*ai*|*vj*), dihitung pada saat *training*. Di mana,

$$P(a_1|v_j) = \frac{|\, n_i + 1 \,|}{|n + kosakata|} \qquad ...(2.5)$$

Keterangan:

1. Doc_j: kumpulan dokumen yang memiliki kategori v_j.

2. |training|: jumlah dokumen yang digunakan dalam pelatihan (kumpulan data latih).

3. n: jumlah total kata yang terdapat di dalam kata tekstual yang memiliki nilai fungsi target yang sesuai.

4. n_i: jumlah kemunculan kata a_i pada semua data tekstual yang memiliki nilai fungsi target yang sesuai.

5. | kosakata |: jumlah kata yang berbeda yang muncul dalam seluruh data tekstual yang digunakan.

Di mana n_i adalah jumlah kemunculan kata a_i dalam dokumen yang berkategor$_i$ v_j, sedangkan *n* adalah banyaknya seluruh kata dalam dokumen dengan kategori v_j dan |*kosakata*| adalah banyaknya kata dalam contoh pelatihan.

2.2.2 Karakteristik Naïve Bayes Classifier

Klasifikasi dengan *Naïve Bayes Classifier* bekerja berdasarkan teori probabilitas yang memandang semua fitur dari data sebagai bukti dalam probabilitas. (F Arfiana, 2014) Hal ini memberikan karakteristik *Naïve Bayes Classifier* sebagai berikut:

1. Metode *Naïve Bayes Classifier* tahan uji (*robust*) terhadap data-data yang terisolasi yang biasanya merupakan data dengan karakteristik berbeda (*outlier*). *Naïve Bayes Classifier* juga bisa menangani nilai atribut yang salah dengan mengabaikan data latih selama proses pembangunan model dan prediksi.
2. Rentan menghadapi atribut yang tidak relevan.
3. Atribut yang mempunyai korelasi bisa mendegradasi kinerja klasifikasi *Naïve Bayes Classifier* karena asumsi independen atribut tersebut sudah tidak ada.

2.2.3 Kelebihan dan Kekurangan Naïve Bayes Classifier

Kelebihan dari penggunaan *Naïve bayes* dalam klasifikasi dokumen dapat ditinjau dari prosesnya yang mengambil aksi berdasarkan data-data yang telah ada sebelumnya. Oleh karena itu, klasifikasi dokumen dengan metode ini dapat dipersonalisasi, maksudnya adalah proses klasifikasi dokumen dapat disesuaikan sesuai dengan sifat dan kebutuhan masing-masing orang.

Keuntungan ini secara nyata diperlihatkan dalam contoh *spam filtering*. Pernyataan suatu surat elektronik adalah spam atau tidak berbeda-beda bergantung

pada subyek pembacanya yang berbeda-beda. Suatu surat elektronik yang diklarifikasikan spam oleh satu orang mungkin diklasifikasikan bukan spam oleh orang lain, dan begitu pula sebaliknya. Dengan klasifikasi metode *Naïve bayes*, pengklasifikasian spam otomatis ini dapat disesuaikan dengan masing-masing orang sehingga meminimalkan aksi salah pengklasifikasian secara personal.

Kekurangan dari metode *Naïve bayes* ini adalah banyaknya celah untuk mengurangi keefektifan metode ini dan akibatnya meloloskan dokumen ke dalam kelas tertentu padahal jelas-jelas dokumen tersebut tidak layak berada di kelas tersebut. Dalam kasus *spam filtering,* kelemahan ini banyak digunakan oleh *spammer* berpengalaman untuk meloloskan spam ke dalam kelas bukan spam (dictio, 2017).

Banyak cara yang dapat dilakukan, misalnya dengan memasukkan kata-kata yang asing dituliskan sehingga perangkat lunak tidak dapat melakukan pengecekan atau memasukkan banyak kata yang sebenarnya sering digunakan oleh surat elektronik non- spam agar pengguna secara manual mendeteksi sebagai spam. Cara lain adalah dengan memanfaatkan media gambar untuk menyampaikan spam. Hal ini didasarkan kepada metode *Naïve bayes classifier* yang dirancang hanya untuk mendeteksi kata-kata dan bukan gambar. Akibatnya, perangkat lunak tidak mampu untuk menganalisis gambar dan akhirnya mengklasifikasikan spam tersebut ke dalam kelas bukan spam.

2.5. PHP

Menurut Kadir (2015) "*PHP* adalah pemrograman *interpreter* yaitu proses penerjemahan baris kode sumber menjadi kode mesin yang dimengerti komputer

secara langsung pada saat baris kode dijalankan". PHP disebut juga pemrograman *Server Side Programming*, hal ini dikarenakan seluruh prosesnya dijalankan pada *server*. *PHP* adalah suatu bahasa dengan hak cipta terbuka atau yang juga dikenal dengan *open source* yaitu pengguna data mengembangkan kode-kode fungsi sesuai kebutuhannya.

Menurut Shalahuddin (2015:22) "*PHP (Perl Hypertext Preprocessor)* adalah bahasa *server-side-scripting* yang menyatu dengan *HTML* untuk membuat halaman web yang dinamis". Dengan menggunakan program *PHP*, sebuah *website* akan lebih interaktif dan dinamis. Kelebihan-kelebihan dari *PHP* yaitu:

1. *PHP* merupakan sebuah bahasa *script* yang tidak melakukan sebuah kompilasi dalam penggunaannya. Tidak seperti halnya bahasa pemrograman aplikasi yang lainnya.
2. *PHP* dapat berjalan pada *web server* yang dirilis oleh Microsoft, seperti *IIS* atau *PWS* juga pada *apache* yang bersifat *open source*.
3. Karena sifatnya yang *open source*, maka perubahan dan perkembangan interpreter pada *PHP* lebih cepat dan mudah, karena banyak *update* dan *developer* yang siap membantu pengembangannya.
4. Jika dilihat dari segi pemahaman, *PHP* memiliki referensi yang begitu banyak sehingga sangat mudah untuk dipahami.

   PHP dapat berjalan pada 3 *operating system*, yaitu: *Linux*, *UNIX*, dan *windows*, dan juga dapat dijalankan secara *runtime* pada suatu *console*.

2.6. MySQL

MySQL adalah *Relational Database Management System* (*RDBMS*) yang didistribusikan secara gratis di bawah lisensi *GPL (General Public License).* Di mana setiap orang bebas untuk menggunakan *MySQL*, namun tidak boleh dijadikan produk turunan yang bersifat *closed source* atau komersial. *MySQL* sebenarnya merupakan turunan salah satu konsep utama dalam *database* sejak lama, yaitu *SQL (Structured Query Language).SQL* adalah sebuah konsep pengoperasian *database*, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis (Kadir, 2013).Keandalan suatu sistem *database* (*DBMS*) dapat diketahui dari cara kerja *optimizer*-nya dalam melakukan proses perintah-perintah *SQL*, yang dibuat oleh *user* maupun program-program aplikasinya. *MySQL* biasanya digunakan atau di-*install* bersamaan dengan *XAMPP* sehingga untuk melihat isi tabel bisa menggunakan *PHPmyAdmin*.

Sebagai *software database* dengan konsep *database* modern, *MySQL* memiliki banyak kelebihan antara lain:

1. *Portability*

   MySQL dapat digunakan dengan stabil tanpa kendala, berarti pada berbagai sistem operasi di antaranya seperti *Windows, Linux, Mac OS X Server, Solaris, Amiga HP-UX* dan masih banyak lagi. *Open source MySQL* didistribusikan secara *open source* di bawah lisensi *GPL*, sehingga dapat memperoleh menggunakannya secara cuma-cuma tanpa dipungut biaya sepeser pun.

2. *Multiuser*

   MySQL dapat digunakan untuk menangani beberapa *user* dalam waktu yang bersamaan tanpa mengalami masalah atau konflik. Hal ini akan memungkinkan sebuah *database* server *MySQL* dapat diakses *client* secara bersamaan dalam waktu yang bersamaan pula.

3. *Performance Tuning*

   MySQL memiliki kecepatan yang cukup menakjubkan dalam menangani *query* sederhana, serta mampu memproses lebih banyak *SQL* persatuan waktu.

4. *Column Types*

   MySQL didukung tipe kolom(tipe data) yang sangat kompleks.

5. *Command dan Functions*

   MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah *SELECT* dan *WHERE* dalam *query*.

6. *Scalability dan Limits*

   Dalam hal batas kemampuan, *MySQL* terbukti mampu menangani *database* dalam skala yang besar dengan jumlah *record* lebih dari 50 juta dan 60 ribu tabel serta 5 miliar baris. Selain itu batas indeks yang dapat ditampung mencapai 32 indeks pada setiap tabelnya.

7. *Interface*

   Sama halnya dengan *software database* lainnya, MySQL memiliki *interface* (antarmuka) terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi *API (Application Programming Interface).*

8. Struktur tabel

   Struktur tabel *MySQL* cukup baik, serta cukup fleksibel. Misalnya ketika menangani *Alter Table*, dibandingkan *database* lainnya semacam *ProgresSQL* ataupun *Oracle*.

2.7. Penelitian Terdahulu

Menurut Santoso dkk (2018). Berita sebagai salah satu jenis informasi yang dibutuhkan dalam kehidupan sehari-hari telah tersedia secara bebas di internet. Situs berita telah melakukan pengelompokan berita berdasarkan topiknya untuk mempermudah pengguna mencari berita yang dibutuhkan. Klasifikasi dokumen telah banyak digunakan untuk membantu pengelompokan berita secara otomatis. Kurang tersedianya data pelatihan yang cukup untuk digunakan komputer membentuk model klasifikasi yang baik sering menjadi kendala dalam implementasi di kasus nyata. Masalah utama dalam pelabelan data pelatihan agar diperoleh jumlah data yang cukup adalah perlunya biaya yang besar dan waktu yang cukup lama. Algoritme *semi-supervised* telah ditawarkan untuk menjawab permasalahan tersebut dengan menggunakan data berlabel dan tak berlabel dalam membentuk model klasifikasi yang dibutuhkan. Makalah ini mengusulkan sistem klasifikasi berita menggunakan *semi-supervised learning* dengan algoritma *Self-Training Naïve Bayes*. Fitur yang digunakan dalam klasifikasi teks ini adalah model *Word2Vec Skip-Gram*. Model ini banyak digunakan untuk merepresentasikan kata dalam penelitian terbaru di bidang linguistik komputasi atau *text mining*. Alasan utama digunakannya *Word2Vec* adalah dapat menambahkan makna semantik dari kata dalam proses klasifikasi. Data yang digunakan dalam klasifikasi ini adalah data

berita bahasa Indonesia dengan jumlah 29.587 dokumen. Hasil percobaan *Self-Training Naïve Bayes* memiliki nilai *F1-Score* terbaik sebesar 94,17%.

Menurut Riani dkk (2018). Berita merupakan salah satu kebutuhan penting bagi masyarakat di berbagai belahan dunia. Melalui berita, masyarakat dapat mengetahui berbagai informasi yang sedang terjadi dimasyarakat seperti ekonomi, politik, kesehatan, kriminal, maupun bencana alam. Kebutuhan informasi yang meningkat setiap harinya membuat beberapa pihak instansi untuk dapat menyajikan berita secara cepat, tepat, terpercaya, dan akurat melalui media cetak maupun elektronik yang dapat dinikmati oleh pembaca berita. Meningkatnya jumlah berita yang didapat setiap harinya mengakibatkan penumpukan data yang besar berupa dokumen teks baik secara *online* maupun *offline.* Sehingga menyulitkan dalam pencarian dan pengklasifikasian dokumen yang sesuai dengan kebutuhan. Untuk mempermudah dalam pengklasifikasian dokumen teks berita Berbahasa Indonesia salah satunya dengan menggunakan metode *NBC* dan *K-Means Clustering*. Di mana perhitungan *NBC* dilakukan tidak secara acak, sedangkan perhitungan untuk *K-Means Clustering* dilakukan secara acak, dalam penelitian ini perhitungan *k-means clustering* dilakukan 5x pada setiap dokumen sehingga hasil akurasinya kurang akurat jika dibandingkan dengan *NBC*. Hasil akurasi tertinggi yang diperoleh dari penelitian pengklasifikasian berita dengan metode *nbc* sebesar 50% dan rata-rata untuk keseluruhannya dari keempat dokumen sebesar 45.33%, sedangkan hasil akurasi tertinggi pengklasifikasian menggunakan metode *k_means* sebesar 100% dan hasil rata-rata keseluruhannya sebesar 53.26% dengan rincian dokumen_0 diperoleh hasil rata-rata sebesar 54%, dokumen 1 hasil rata-rata sebesar

60%, dokumen 2 hasil rata-rata keseluruhannya sebesar 42.56%, dan dokumen 3 hasil rata-rata keseluruhannya sebesar 52.48%.

Menurut Pramudita dkk (2018), dokumen berita olahraga dalam bentuk web kini memiliki jumlah yang besar dalam kurun waktu singkat. Untuk kemudahan akses dokumen perlu melakukan pengelompokan dokumen berita ke dalam beberapa kategori. Hal tersebut bertujuan agar berita olahraga tersusun sesuai dengan kategori yang ditentukan. Berita dapat dikelompokkan secara manual oleh manusia, akan tetapi hal tersebut membutuhkan waktu yang lama untuk melakukan kategorisasi. Metode klasifikasi diusulkan dalam penelitian ini untuk melakukan pengkategorian secara otomatis dokumen berita. Tujuan dilakukannya klasifikasi adalah untuk mempercepat dan mempermudah dalam pemberian kategori, sehingga dapat meningkatkan efisiensi waktu. Pada penelitian ini menggunakan metode klasifikasi *Naïve Bayes Classifier*. Sebelum dilakukan klasifikasi ada proses *pre-processing* dengan menggunakan *Enhanced Confix Striping Stemmer*. Hal ini bertujuan untuk mengembalikan ke bentuk kata dasar, sehingga data berkurang dan proses komputasi menjadi lebih efisien. Pengujian dilakukan menggunakan 18 berita olahraga yang dipilih secara acak oleh *user* atau tester, dari 18 berita yang diujikan terdapat 14 berita yang bernilai benar atau relevan dengan analisis yang dilakukan use atau tester pada berita uji. Dari penelitian ini dapat disimpulkan bahwa Aplikasi Klasifikasi Berita Olahraga menggunakan Metode *Naïve Bayes* dengan *Enhanced Confix Striping Stemmer* mampu mengklasifikasi berita olahraga sesuai dengan kategori masing-masing, seperti Sepak Bola, Basket, Raket, Formula 1, Moto GP dan olahraga lainnya dengan keakuratan sebesar 77%.