**BAB II**

**LANDASAN TEORI**

1. **Penelitian Terdahulu**

Penelitian Terdahulu dalam sebuah penelitian dapat memberikan kontribusi untuk mengembangkan ilmu pengetahuan. Oleh karena itu, dalam penelitian ini diperlukan beberapa penelitian terdahulu untuk mengumpulkan literasi yang dapat memperkaya referensi dalam melakukan penelitian ini. Adapun penjelasan mengenai kelima penelitian tersebut disajikan dalam Tabel 2. 1 di bawah ini:

**Tabel 2. 1 Penelitian Terdahulu**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Judul Penelitian** | **Nama Peneliti (Tahun)** | **Metode** | **Hasil** |
| 1. | 20 years of research on virtual reality and augmented reality in tourism context: A text-mining approach | Sandra Maria Correia Loureiro, Joao Guerreiro, Faizan Ali (2020) | Text mining | Temuan mengungkapkan 10 topik inti dalam makalah jurnal dan 11 topik inti dalam prosiding konferensi, yang disajikan bersama dengan tinjauan umum studi yang diterbitkan dan penulis utama. |
| 2. | Sistem Aplikasi Penilaian Jawaban Essay Test Calon Karyawan PT Siloam Hospitals TB Simatupang Menggunakan Algoritma Text Mining TF-IDF Berbasis Web | Agung Daniel Sipayung, Fauziah, Nurhayati (2020) | Text mining TD-IDF | Sistem penilaian essay test menggunakan metode text mining sudah bekerja dengan baik dalam penerapan percobaan ujian esai memperlihatkan kecocokan penilaian pada aplikasi dengan yang nilai dibuat oleh HRD dengan average 73,51%. |
| 3. | IMPLEMENTASI TEXT MINING PADA TWITTER DENGAN ALGORITMA K-MEANS CLUSTERING SEBAGAI DASAR KEBIJAKAN MARKETING BIRO PERJALANAN WISATA | Hendry Fonda, Budy Mustika, Eka Sabna, Yuda Irawan (2020) | Deskriptif analitis | Hasil yang diperoleh dari pengelompokan data tersebut menghasilkan data berupa kata-kata yang sedang tren dibicarakan. Hasil inilah yang digunakan oleh pelaku usaha untuk menentukan arah kebijakan pemasaran produknya. Hasil tersebut juga bisa dimanfaatkan oleh pelaku usaha sebagai bahan prediksi terhadap kecenderungan masyarakat dalam berwisata. |
| 4. |  |  |  |  |
| 5. |  |  |  |  |
| 6. |  |  |  |  |
| 7. |  |  |  |  |
| 8. |  |  |  |  |
| 9. |  |  |  |  |
| 10. |  |  |  |  |

1. **Text Mining**

*Text Mining* (Penambangan Teks), juga disebut sebagai *Text Data Mining* adalah proses memperoleh informasi dari data teks yang sebelumnya tidak diketahui dan tidak segera terungkap. Ini melibatkan transformasi teks menjadi data numerik yang dapat digunakan dalam algoritma penambangan data(Zhang dkk., 2019). Karakteristik *real-time* dan penting dari media sosial, seperti Twitter, dapat memberikan peluang besar untuk dapat memahami gambaran umum opini publik terhadap suatu topik. Pada penelitian ini, pendekatan text mining digunakan untuk membantu memahami gambaran umum opini publik terhadap kenaikan harga BBM. *Text Mining* adalah perpanjangan dari teknik *Data Mining*, dan Text Mining bergantung pada representasi dokumen berbasis fitur yang dapat mencakup kata, konsep, karakter, dan Istilah. Teknik text mining mencakup proses penyuntingan, pengorganisasian, dan analisis sejumlah besar dokumen untuk memberikan informasi spesifik kepada peneliti akademis dan mengidentifikasi hubungan yang diidentifikasi oleh informasi tersebut. Masing-masing teknik ini dapat digunakan dalam menemukan beberapa informasi non-sepele dari kumpulan dokumen. Langkah-langkah dari proses text mining meliputi pemilihan data, pembersihan data, penguraian teks, penyaringan teks, dan interpretasi hasil(Shen dkk., 2019).

1. **Natural Language Processing (NLP)**

Bidang Natural Language Processing, juga dikenal sebagai linguistik komputasi, melibatkan rekayasa model komputasi dan proses untuk memecahkan masalah praktis dalam memahami bahasa manusia. Pekerjaan di NLP dapat dibagi menjadi dua subarea yang luas: area inti dan aplikasi, meskipun terkadang sulit untuk membedakan dengan jelas area mana yang menjadi masalah. Area inti mengatasi masalah mendasar seperti pemodelan bahasa, yang menggarisbawahi asosiasi kuantifikasi di antara kata-kata yang muncul secara alami; pemrosesan morfologi, berurusan dengan segmentasi komponen kata yang bermakna dan mengidentifikasi bagian kata yang sebenarnya (POS) dari kata-kata yang digunakan; pemrosesan sintaksis, atau penguraian, yang membangun diagram kalimat sebagai kemungkinan pendahulu untuk pemrosesan semantik; dan pemrosesan semantik, yang berupaya menyaring makna kata, frasa, dan komponen tingkat yang lebih tinggi dalam teks. Area aplikasi melibatkan topik, seperti ekstraksi informasi yang berguna (misalnya, entitas dan relasi bernama), terjemahan teks antara dan di antara bahasa, ringkasan karya tulis, jawaban otomatis pertanyaan dengan menyimpulkan jawaban, dan klasifikasi dan pengelompokan dokumen(Otter dkk., 2021). Seringkali, seseorang perlu menangani satu atau lebih masalah inti dengan sukses dan menerapkan ide dan prosedur tersebut untuk memecahkan masalah praktis. Saat ini, NLP terutama merupakan bidang berbasis data yang menggunakan komputasi statistik dan probabilistik bersama dengan pembelajaran mesin contohnya *Naïve Bayes, k-nearest neighbors, hidden Markov models, conditional random fields (CRFs), decision trees, random forests,* dan *support vector machines*, banyak digunakan.

* 1. **Analisis Sentimen**

Salah satu bidang penelitian paling produktif dalam ilmu komputer adalah analisis sentimen, yang bertujuan untuk mengidentifikasi dan mengekstrak opini pengguna. Penelitian analisis sentimen sangat berarti untuk mendapatkan tren sentimen dari komentar. Analisis sentimen adalah tugas NLP yang minatnya telah meningkat dalam komunitas ilmiah karena banyak tantangan terbuka yang menarik, serta dunia bisnis karena manfaat luar biasa yang bisa didapat dari prediksi pemasaran dan keuangan. Walaupun sebagian besar karya mendekatinya sebagai masalah kategorisasi sederhana, analisis sentimen sebenarnya adalah masalah penelitian kompleks yang membutuhkan penanganan banyak tugas NLP, termasuk deteksi subjektivitas, resolusi anaphora, disambiguation arti kata, deteksi sarkasme, ekstraksi aspek, dan banyak lagi(Cambria dkk., 2020). Analisis sentimen adalah sejenis klasifikasi teks, yang melibatkan pemrosesan bahasa alami, pembelajaran mesin, penambangan data, pencarian informasi, dan bidang penelitian lainnya. Analisis sentimen komentar terutama berfokus pada analisis orientasi sentimen dari komentar, yang menunjukkan bahwa pengguna mengekspresikan sentimen positif, negatif, atau netral terhadap produk atau peristiwa. Selain itu, analisis sentimen dapat dibagi menjadi analisis komentar berita, analisis komentar produk, analisis komentar film, dan jenis lainnya. Komentar ini menyampaikan pandangan pengguna Internet tentang produk, acara hangat, dll. Pedagang dapat menguasai kepuasan pengguna dengan komentar produk yang relevan. Pengguna potensial dapat mengevaluasi produk dengan melihat komentar produk ini(Xu dkk., 2019).

* 1. **Naïve Bayes**

Klasifikasi adalah salah satu masalah mendasar dalam penambangan data, di mana algoritma klasifikasi mencoba membangun pengklasifikasi dari serangkaian contoh pelatihan yang diberikan dengan label kelas. Karena kemudahannya untuk di konstruksi dan diinterpretasikan, bersama dengan kinerjanya yang baik, Naïve Bayes (NB) banyak digunakan untuk mengatasi masalah klasifikasi dalam aplikasi dunia nyata(Jiang dkk., 2019). Keuntungan utama dari Naïve Bayes Classifier adalah sangat mudah dibuat dan tidak memerlukan skema estimasi parameter *iterative* yang rumit. Selain itu, Naïve Bayes Classifier tahan terhadap noise dan atribut yang tidak relevan(He dkk., 2019). Naïve Bayes Classifier adalah metode klasifikasi berdasarkan teorema Bayes. Naïve Bayes Classifier dikenal lebih baik dari beberapa metode klasifikasi lainnya. Karena pertama, ciri utama Naïve Bayes adalah asumsi independensi (naif) yang sangat kuat dari setiap kondisi atau peristiwa. Kedua, model nya sederhana dan mudah dibuat. Ketiga, model dapat diimplementasikan untuk kumpulan data yang besar(Salmi & Rustam, 2019). Dasar dari teorema Naïve Bayes yang digunakan adalah rumus Bayes seperti pada persamaan ( 1 )

|  |  |
| --- | --- |
|  | ( 1 ) |

Keterangan:

Peluang kejadian A sebagai B ditentukan dari peluang B saat A, peluang A, dan peluang B. Pada pengaplikasian nya nanti rumus pada persamaan ( 1 ) berubah menjadi persamaan ( 2 )

|  |  |
| --- | --- |
|  | ( 2 ) |

Naïve Bayes atau bisa disebut sebagai Multinomial Naïve Bayes merupakan model penyederhanaan dari Metoda Bayes yang cocok pengklasifikasian teks atau dokumen.

Pada persamaan ( 3 ) merupakan persamaan model penyederhanaan dari Metoda Bayes.

|  |  |
| --- | --- |
|  | ( 3 ) |

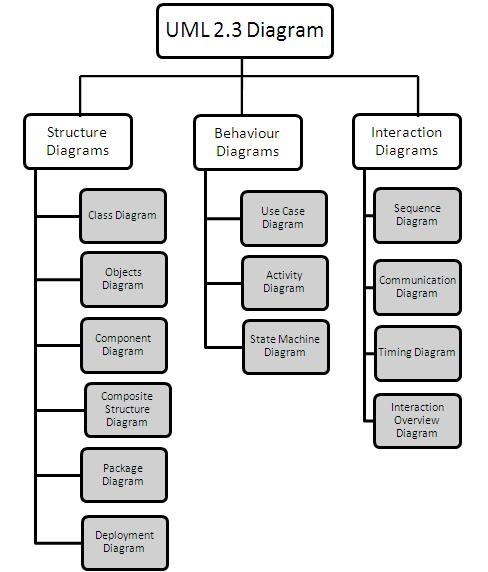
Berdasarkan persamaan ( 3 ), maka persamaan ( 1 ) dapat ditulis seperti yang terdapat pada persamaan ( 4 )

|  |  |
| --- | --- |
|  | ( 4 ) |

1. **UML**

Pada perkembangan teknik pemrograman berorientasi objek, muncullah sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemrograman berorientasi objek, yaitu Unified Modeling Language (UML). UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak. UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung. Secara fisik, UML adalah sekumpulan spesifikasi yang dikeluarkan oleh OMG. UML terbaru adalah UML 2.3 yang terdiri dari 4 macam spesifikasi, yaitu *Diagram Interchange Specification*, UML Infrastructure, UML Superstructure, dan Object Constraint Language (OCL)(Rosa A. S. & M. Shalahuddin, 2018).

Pada UML 2.3 terdiri dari 13 macam diagram yang dikelompokkan dalam 3 kategori. Pembagian kategori dan macam-macam diagram tersebut dapat dilihat pada Gambar 2. 1 di bawah.



**Gambar 2. 1 Diagram UML**

Berikut ini penjelasan singkat dari pembagian kategori tersebut.

1. *Structure diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan.
2. *Behavior diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem.
3. *Interaction diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun interaksi antar subsistem pada suatu sistem(Rosa A. S. & M. Shalahuddin, 2018).
4. **Class Diagram**

Diagram kelas atau class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

1. Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas.
2. Operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas.

Diagram kelas dibuat agar pembuat program atau *programmer* membuat kelas-kelas sesuai rancangan di dalam diagram kelas agar antara dokumentasi perancangan dan perangkat lunak sinkron. Banyak berbagai kasus, perancangan kelas yang dibuat tidak sesuai dengan kelas-kelas yang dibuat pada perangkat lunak, sehingga tidaklah ada gunanya lagi sebuah perancangan karena apa yang dirancang dan hasil jadinya tidak sesuai.

Kelas-kelas yang ada pada struktur sistem harus dapat melakukan fungsi-fungsi sesuai dengan kebutuhan sistem sehingga pembuat perangkat lunak atau programmer dapat membuat kelas-kelas di dalam program perangkat lunak sesuai dengan perancangan diagram kelas(Rosa A. S. & M. Shalahuddin, 2018). Adapun simbol-simbol class diagram dapat dilihat pada Tabel 2. 2

**Tabel 2. 2 Simbol pada Class Diagram**

|  |  |
| --- | --- |
| **Simbol** | **Deskripsi** |
| Kelas | kelas pada struktur sistem |
| antarmuka/*interface*  **nama\_interface** | sama dengan konsep *interface* dalam pemrograman berorientasi objek |
| asosiasi/*association* | relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai dengan *multiplicity* |
| asosiasi berarah/ *directed association* | relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan *multiplicity* |
| generalisasi | relasi antarkelas dengan makna generalisasi-spesialisasi (umum khusus) |
| kebergantungan/*dependency* | relasi antarkelas dengan makna kebergantungan antarkelas |
| agregasi/*aggregation* | relasi antar kelas dengan makna semua-bagian (*whole-part*) |

*Sumber:* (Rosa A. S. & M. Shalahuddin, 2018)

1. **Use Case Diagram**

*Use case* atau diagram *use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. Use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu(Rosa A. S. & M. Shalahuddin, 2018). Adapun simbol-simbol *use case diagram* dapat dilihat pada Tabel 2. 3.

**Tabel 2. 3 Simbol pada Use Case Diagram**

|  |  |
| --- | --- |
| **Simbol** | **Deskripsi** |
| *Use Case*  **nama use case** | fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal di awal frase nama *use case* |
| Aktor / *actor*  Leveling Use Case – Ini WordPress ?  **nama aktor** | orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor |
| Asosiasi / *association* | Komunikasi antara aktor dan *use case* yang berpartisipasi pada *use case* atau *use case* memiliki interaksi dengan aktor |
| Ekstensi / *extend*  <<extend>> | relasi *use case* tambahan ke sebuah *use case* dimana *use case* yang ditambahkan dapat berdiri sendiri walau tanpa *use case* tambahan itu; mirip dengan prinsip *inheritance* pada pemrograman berorientasi objek; biasanya *use case* tambahan memiliki nama depan yang sama dengan *use case* yang ditambahkan, misal    arah panah mengarah pada *use case* yang ditambahkan; biasanya *use case* yang menjadi *extend-*nya merupakan jenis yang sama dengan *use case* yang menjadi induknya |
| Generalisasi / *generalization* | hubungan generalisasi dan spesialisasi (umum – khusus) antara dua buah *use case* dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya, misalnya:    arah panah mengarah pada *use case* yang menjadi generalisasi nya (umum) |
| Menggunakan / *include* / *uses*  <<include>>    <<uses>> | relasi *use case* tambahan ke sebuah *use case* di mana *use case* yang ditambahkan memerlukan *use case* ini untuk menjalankan fungsinya atau sebagai syarat dijalankan *use case* ini  ada dua sudut pandang yang cukup besar mengenai include di *use case:*   * *Include* berarti *use case* yang ditambahkan akan selalu dipanggil saat *use case* tambahan dijalankan, misal pada kasus berikut:      * *Include* berarti *use case* yang tambahan akan selalu melakukan pengecekan apakah *use case* yang ditambahkan telah dijalankan sebelum *use case* tambahan dijalankan, misal pada kasus berikut:     Kedua interpretasi di atas dapat dianut salah satu atau keduanya tergantung pada pertimbangan dan interpretasi yang dibutuhkan. |

*Sumber:* (Rosa A. S. & M. Shalahuddin, 2018)

1. **Activity Diagram**

Diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem(Rosa A. S. & M. Shalahuddin, 2018). Adapun simbol-simbol activity *diagram* dapat dilihat pada Tabel 2. 4

**Tabel 2. 4 Simbol pada Activity Diagram**

|  |  |
| --- | --- |
| **Simbol** | **Deskripsi** |
| status awal | status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal |
| aktivitas | aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja |
| percabangan / *decision* | asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu |
| penggabungan / *join* | asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu |
| status akhir | status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir |
| swimlane    atau | memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi |

*Sumber:* (Rosa A. S. & M. Shalahuddin, 2018)

1. **Sequence Diagram**

Diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambar diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah use case beserta metode-metode yang dimiliki kelas yang *diinstansiasi* menjadi objek itu. Membuat diagram sekuen juga dibutuhkan untuk melihat skenario yang ada pada *use case*(Rosa A. S. & M. Shalahuddin, 2018). Adapun simbol-simbol *sequence diagram* dapat dilihat pada Tabel 2. 5.

**Tabel 2. 5 Simbol pada Sequence Diagram**

|  |  |
| --- | --- |
| **Simbol** | **Deskripsi** |
| Aktor  **Description: E:\index.png**  **nama aktor**  atau  nama aktor  tanpa waktu aktif | orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor |
| Garis hidup / *lifeline* | menyatakan kehidupan suatu objek |
| Objek  nama objek: nama kelas | menyatakan objek yang berinteraksi pesan |
| Waktu aktif | menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya, misalnya    maka cekStatusLogin () dan open () dilakukan di dalam metode login ()  Aktor tidak memiliki waktu aktif |
| Pesan tipe *create*  <<*create*>> | menyatakan suatu objek membuat objek yang lain, arah panah mengarahkan pada objek yang dibuat |
| Pesan tipe *call*  1: *nama*\_metode() | menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri,    arah panah mengarah pada objek yang memiliki operasi/metode, karena ini memanggil operasi/metode maka operasi/metode yang dipanggil harus ada pada diagram kelas sesuai dengan kelas objek yang berinteraksi |
| Pesan tipe *send*  1: masukan | menyatakan bahwa suatu objek mengirimkan data/masukkan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirimi |
| Pesan tipe *return*  1: keluaran | menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian |
| Pesan tipe *destroy* | menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada create maka ada *destroy* |

*Sumber: (Rosa A. S. & M. Shalahuddin, 2018)*

**DAFTAR PUSTAKA**

Cambria, E., Li, Y., Xing, F. Z., Poria, S., & Kwok, K. (2020). SenticNet 6: Ensemble Application of Symbolic and Subsymbolic AI for Sentiment Analysis. *International Conference on Information and Knowledge Management, Proceedings*, 105–114. https://doi.org/10.1145/3340531.3412003

He, Q., Shahabi, H., Shirzadi, A., Li, S., Chen, W., Wang, N., Chai, H., Bian, H., Ma, J., Chen, Y., Wang, X., Chapi, K., & Ahmad, B. bin. (2019). Landslide spatial modelling using novel bivariate statistical based Naïve Bayes, RBF Classifier, and RBF Network machine learning algorithms. *Science of the Total Environment*, *663*, 1–15. https://doi.org/10.1016/j.scitotenv.2019.01.329

Jiang, L., Zhang, L., Yu, L., & Wang, D. (2019). Class-specific attribute weighted naive Bayes. *Pattern Recognition*, *88*, 321–330. https://doi.org/10.1016/j.patcog.2018.11.032

Otter, D. W., Medina, J. R., & Kalita, J. K. (2021). A Survey of the Usages of Deep Learning for Natural Language Processing. *IEEE Transactions on Neural Networks and Learning Systems*, *32*(2), 604–624. https://doi.org/10.1109/TNNLS.2020.2979670

Rosa A. S., & M. Shalahuddin. (2018). *Rekayasa Perangkat Lunak: Terstruktur dan Berorientasi Objek Edisi Revisi*.

Salmi, N., & Rustam, Z. (2019). Naïve Bayes Classifier Models for Predicting the Colon Cancer. *IOP Conference Series: Materials Science and Engineering*, *546*(5). https://doi.org/10.1088/1757-899X/546/5/052068

Shen, C. wen, Min Chen, & Wang, C. chen. (2019). Analyzing the trend of O2O commerce by bilingual text mining on social media. *Computers in Human Behavior*, *101*, 474–483. https://doi.org/10.1016/j.chb.2018.09.031

Xu, G., Meng, Y., Qiu, X., Yu, Z., & Wu, X. (2019). Sentiment analysis of comment texts based on BiLSTM. *IEEE Access*, *7*, 51522–51532. https://doi.org/10.1109/ACCESS.2019.2909919

Zhang, F., Fleyeh, H., Wang, X., & Lu, M. (2019). Construction site accident analysis using text mining and natural language processing techniques. *Automation in Construction*, *99*, 238–248. https://doi.org/10.1016/j.autcon.2018.12.016