

# IDENTIFIKASI JENIS OPERASI DATA MANIPULATION LANGUAGE BERBASIS BILSTM PADA KALIMAT BERBAHASA INDONESIA

Agung Prasetya <sup>\*1)</sup>, Yayak Kartika Sari <sup>2)</sup>, Joko Iskandar <sup>3)</sup>, Mohamad Khoirul Ansor <sup>4)</sup>

1. Program Studi Informatika, Fakultas Sains dan Teknologi, Universitas Bhinneka PGRI, Indonesia
2. Program Studi Informatika, Fakultas Sains dan Teknologi, Universitas Bhinneka PGRI, Indonesia
3. Program Studi Informatika, Fakultas Sains dan Teknologi, Universitas Bhinneka PGRI, Indonesia
4. Program Studi Informatika, Fakultas Sains dan Teknologi, Universitas Bhinneka PGRI, Indonesia

## Article Info

**Kata Kunci:** Bidirectional Long Short-Term Memory; BiLSTM; Data Manipulation Language; Text-to-SQL; Word Embedding.

**Keywords:** Bidirectional Long Short-Term Memory; BiLSTM; Data Manipulation Language; Text-to-SQL; Word Embedding.

## Article history:

Received 20 September 2024

Revised 19 Oktober 2024

Accepted 25 November 2024

Available online 1 December 2024

## DOI :

<https://doi.org/10.29100/jupi.v9i4.8695>

\*Corresponding author.

Agung Prasetya

E-mail address:

[agung@ubhi.ac.id](mailto:agung@ubhi.ac.id)

## ABSTRAK

Text-to-SQL memungkinkan penggunaan bahasa alami untuk mendapatkan informasi dari database. Melalui pendekatan ini, pengguna non teknis tidak perlu memahami sintaks SQL untuk melakukan query database. Penelitian ini mengusulkan pendekatan berbasis Bidirectional Long Short-Term Memory (BiLSTM) untuk mengklasifikasikan jenis operasi Data Manipulation Language (DML), seperti SELECT, INSERT, UPDATE, dan DELETE, pada kalimat berbahasa Indonesia. Pendekatan dibangun dengan merepresentasikan kalimat sebagai urutan vektor kata menggunakan word embedding, lalu diproses oleh arsitektur BiLSTM untuk menangkap konteks sekuensial dua arah. Dataset berisi 1600 kalimat dari tiga domain utama: pendidikan, e-commerce, dan layanan publik. Setiap kalimat telah dianotasi sesuai dengan operasi DML yang terkandung. Hasil evaluasi menunjukkan bahwa model BiLSTM mencapai akurasi sebesar 93% dan F1-score sebesar 92%. Analisis per label mengungkapkan bahwa model sangat efektif mengenali operasi SELECT dan INSERT, namun sedikit kesulitan membedakan UPDATE dan DELETE. Penelitian ini menunjukkan bahwa pendekatan BiLSTM mampu mengklasifikasikan tipe DML secara efektif dan efisien dalam konteks bahasa Indonesia.

## ABSTRACT

Text-to-SQL enables the use of natural language to obtain information from a database. Through this approach, non-technical users do not need to understand SQL syntax to query the database. This research proposes a Bidirectional Long Short-Term Memory (BiLSTM) based approach to classify the types of Data Manipulation Language (DML) operations, such as SELECT, INSERT, UPDATE, and DELETE, on Indonesian sentences. The approach is built by representing sentences as a sequence of word vectors using word embedding, and then processed by the BiLSTM architecture to capture bidirectional sequential context. The dataset contains 1600 sentences from three main domains: education, e-commerce, and public services. Each sentence has been annotated according to the DML operations contained. The evaluation results show that the BiLSTM model achieves an accuracy of 93% and an F1-score of 92%. Per-label analysis revealed that the model was very effective at recognizing SELECT and INSERT operations, but had little difficulty distinguishing UPDATE and DELETE. This research shows that the BiLSTM approach is able to classify DML types effectively and efficiently in the Indonesian context.

## I. PENDAHULUAN

Text-to-SQL memungkinkan penggunaan bahasa alami untuk mendapatkan informasi dari database [1], [2], [3], [4]. Melalui pendekatan ini, pengguna non teknis tidak perlu memahami sintaks SQL untuk melakukan query database. Namun, menerjemahkan teks berbahasa alami ke query bukanlah tugas yang sederhana. Bahasa alami memiliki struktur yang cenderung bebas dan ambiguitas serta variasi ekspresi yang luas. Semua ini menyulitkan sistem untuk menangkap maksud pengguna. Selain itu, query memiliki struktur sintaksis yang kaku

dan ketat sehingga kesalahan kecil dalam interpretasi berpotensi menghasilkan query yang salah.

Sejumlah pendekatan telah diusulkan untuk menyelesaikan tugas Text-to-SQL. Pendekatan berbasis aturan (rule-based) [5], [6], [7] menggunakan seperangkat aturan linguistik dan sintaksis yang merelasikan struktur kalimat ke bentuk query. Untuk mengatasi kompleksitas query, pendekatan berbasis template [8], [9], [10] berupaya mencocokkan entitas dan kata kunci yang terkandung di kalimat ke pola-pola query yang telah ditentukan sebelumnya. Pendekatan ini lebih mampu menangani variasi query yang lebih kompleks. Pendekatan berbasis encoder-decoder [11], [12], [13] diusulkan untuk mengurangi pembentukan template setiap kali ada pola query baru. Pendekatan ini memodelkan proses translasi kalimat ke query sebagai masalah sequence-to-sequence. Untuk mengatasi adanya teks yang berada di luar distribusi data latih atau kosa kata yang belum terlihat sebelumnya, pendekatan berbasis Large Language Models (LLM) [14], [15], [16], [17] diusulkan dalam text-to-SQL. LLM telah dilatih pada korpus besar dengan beragram semantik sehingga memiliki kemampuan generalisasi yang lebih baik.

Meskipun berbagai pendekatan tersebut berhasil menerjemahkan kalimat alami menjadi query SQL, sebagian besar tidak secara eksplisit mendeteksi jenis operasi Data Manipulation Language (DML) seperti SELECT, INSERT, UPDATE, atau DELETE. Model-model tersebut langsung langsung menerjemahkan teks ke query akhir tanpa tahap identifikasi jenis operasi terlebih dahulu. Tidak adanya identifikasi eksplisit terhadap jenis operasi dapat menyebabkan kegagalan sistem text-to-SQL untuk membedakan antara permintaan untuk mengambil data dan permintaan untuk memodifikasi data. Akibatnya, query yang dihasilkan berpotensi tidak sesuai dengan maksud pengguna. Query yang dihasilkan berpotensi pula membahayakan integritas data apabila sistem secara keliru mengeksekusi tindakan destruktif seperti DELETE atau UPDATE. Selain itu, tanpa adanya informasi awal tentang jenis operasi, ruang pencarian query menjadi lebih luas dan tidak terarah. Hal ini dapat menurunkan akurasi dan efisiensi proses pembentukan query.

Untuk mengatasi permasalahan tersebut, penelitian ini mengusulkan masalah dalam identifikasi jenis operasi DML dari suatu kalimat sebagai masalah klasifikasi. Masalah ini diselesaikan dengan pendekatan Bidirectional Long Short-Term Memory (BiLSTM) [18], [19], [20], [21]. BiLSTM dipilih karena mampu menangkap konteks kata dari dua arah. Konteks dua arah ini penting untuk memahami struktur dan makna kalimat secara lebih menyeluruh. Pendekatan ini sesuai untuk menangani fleksibilitas susunan kata dan variasi ekspresi. Output dari BiLSTM kemudian diteruskan ke classifier untuk memetakan representasi kalimat ke salah satu jenis operasi DML, seperti SELECT, INSERT, UPDATE, atau DELETE. Adanya pemisahan proses identifikasi operasi DML dari tahap translasi, sistem dapat membatasi ruang pencarian query sehingga meningkatkan efisiensi dan akurasi dalam proses pembentukan query.

Berbeda dari penelitian-penelitian terdahulu yang menggunakan teks berbahasa Inggris, pendekatan usulan dalam penelitian ini difokuskan pada kalimat-kalimat berbahasa Indonesia. Adanya perbedaan dalam struktur kalimat, urutan kata, dan penggunaan istilah membuat pengujian pendekatan usulan dalam konteks bahasa Indonesia menjadi penting. Untuk menilai kemampuan generalisasi dan akurasi model, pengujian dilakukan pada beberapa skema database dari beberapa domain.

## II. METODE PENELITIAN

Penyelesaian text-to-SQL paling awal menggunakan pendekatan berbasis rule. Pendekatan ini menggunakan aturan linguistik untuk menghubungkan struktur kalimat dengan bentuk query. Salah satu contoh awal dari pendekatan ini adalah sistem LUNAR dan MASQUE/SQL, yang memanfaatkan analisis sintaksis dan semantik [5], [6], [7], [22], [23], [24]. Pendekatan ini cukup efektif dalam domain terbatas. Namun pendekatan ini kurang fleksibel dan tidak optimal menangani bahasa alami yang kompleks serta sulit diperluas ke domain baru.

Untuk mengatasi keterbatasan fleksibilitas rule-based, pendekatan template-based [5], [6], [7], [25], [26], [27] mulai diusulkan. Dalam pendekatan ini, query dibentuk dengan mencocokkan komponen dalam kalimat (seperti entitas dan atribut) ke dalam pola-pola query yang telah ditentukan sebelumnya. Meskipun lebih fleksibel dibanding rule-based, pendekatan template-based tetap memiliki keterbatasan karena perlu merancang template baru untuk setiap pola query yang berbeda. Hal ini membuat pendekatan ini tidak efisien untuk diterapkan dalam domain yang sangat beragam.

Kemudian, pendekatan berbasis encoder-decoder diperkenalkan dengan memodelkan Text-to-SQL sebagai permasalahan sequence-to-sequence [28], [29], [30]. Model seperti Seq2SQL dan SQLNet memanfaatkan representasi semantik teks dan skema database untuk menghasilkan query secara end-to-end melalui proses encoding dan decoding. Pendekatan ini mengurangi ketergantungan terhadap pembuatan template dengan melatih model dataset. Namun demikian, pendekatan ini masih memiliki keterbatasan dalam mengatasi kalimat yang berada di luar distribusi data pelatihan atau mengandung kosa kata yang tidak dikenal sebelumnya (out-of-vocabulary).

Untuk mengatasi permasalahan tersebut, penelitian terkini memanfaatkan Large Language Models (LLM) seperti T5, Codex, dan ChatGPT, yang telah dilatih menggunakan korpus besar dan beragam [14], [15], [16], [17]. LLM memiliki kemampuan generalisasi dan dapat memahami struktur semantik serta konteks secara lebih baik. Beberapa studi menunjukkan bahwa LLM mampu menghasilkan query yang kompleks tanpa pelatihan dengan dataset tertentu. Meskipun demikian, LLM memiliki kekurangan yaitu kecenderungan menghasilkan query yang tidak valid jika tidak dibatasi oleh langkah pra-pemrosesan yang tepat.

Sebagian besar pendekatan-pendekatan tersebut secara langsung memetakan teks ke query secara menyeluruh tanpa terlebih dahulu mengidentifikasi jenis operasi DML yang terkandung dalam kalimat. Padahal, identifikasi jenis operasi seperti SELECT, INSERT, UPDATE, dan DELETE dapat berperan penting dalam menyempitkan ruang pencarian dan mengarahkan model ke struktur query yang lebih tepat.

Dalam konteks bahasa Indonesia, pendekatan Text-to-SQL masih terbatas. Sebagian besar dataset dan model dilatih menggunakan teks berbahasa Inggris sehingga belum banyak pendekatan yang diuji untuk struktur dan ekspresi bahasa Indonesia. Padahal, perbedaan karakteristik linguistik dapat memengaruhi efektivitas pendekatan yang diadopsi dari bahasa lain.

Penelitian ini mengusulkan proses identifikasi jenis operasi DML sebagai masalah klasifikasi. Pada masalah ini, setiap kata dalam kalimat masukan dipetakan ke vektor semantik kata. Vektor-vektor ini kemudian dikodekan menggunakan Bi-LSTM menghasilkan vektor semantik kalimat. Hasil pengkodean ini kemudian menjadi input untuk classifier softmax. Classifier menghasilkan tipe operasi Data Manipulation Language (DML).

Secara keseluruhan sistem bertujuan mengklasifikasikan kalimat ke dalam salah satu dari empat jenis operasi DML: SELECT, INSERT, UPDATE, dan DELETE. Kombinasi BiLSTM dan neural classifier dipilih karena mampu menangkap pola sekuensial dan hubungan semantik antar kata secara kontekstual. Pendekatan usulan memiliki arsitektur yang terdiri dari empat tahap utama: prapemrosesan data, representasi embedding, encoding sekuens menggunakan BiLSTM, dan klasifikasi akhir melalui multilayer perceptron (MLP).

Tahapan awal adalah preprocessing data teks. Simbol atau karakter non-alfabetik dihilangkan dari setiap kalimat. Kalimat kemudian dilakukan tokenisasi menggunakan tokenizer bahasa Indonesia. Setelah itu, token-token diubah ke vektor semantik menggunakan pretrained word embedding FastText untuk bahasa Indonesia. Word embedding ini memungkinkan kata-kata yang bermakna serupa memiliki representasi vektor yang dekat dalam ruang vektor, serta mampu menangani kata tidak dikenal melalui teknik subword. Satu kalimat kemudian dipetakan menjadi satu vektor berukuran tetap dengan padding nol di bagian akhir jika panjang kalimat kurang dari maksimum T token.

Vektor embedding dari tiap kata kemudian dikirimkan sebagai input ke layer BiLSTM. Layer ini terdiri dari dua unit LSTM yang berjalan dari arah kiri ke kanan (LSTM)<sup>→</sup> dan dari kanan ke kiri (LSTM)<sup>←</sup>. Hidden state dari masing-masing arah pada posisi waktu t dinotasikan sebagai (h<sub>t</sub>)<sup>→</sup> dan (h<sub>t</sub>)<sup>←</sup>, kemudian digabungkan sebagai:

$$h_t = [\overrightarrow{LSTM}; \overleftarrow{LSTM}] \in \mathbb{R}^{2h} \quad (1)$$

Untuk mendapatkan representasi kalimat secara keseluruhan, digunakan teknik agregasi temporal berupa max pooling atau average pooling:

$$v = \text{MaxPool}(h_1, h_2, \dots, h_T) \quad (2)$$

atau

$$v = (1/T) \times \sum_t h_t \quad (3)$$

Vektor  $v \in \mathbb{R}^{2h}$  ini selanjutnya menjadi input bagi modul klasifikasi.

Modul klasifikasi terdiri dari dua layer dense non-linier dan satu layer output. Layer pertama mengubah vektor input  $v$  menjadi representasi laten menggunakan fungsi aktivasi ReLU:

$$z_1 = \text{ReLU}(W_1 \times v + b_1) \quad (4)$$

Selanjutnya, representasi  $z_1$  diubah kembali dengan layer kedua:

$$z_2 = \text{ReLU}(W_2 \times z_1 + b_2) \quad (5)$$

Layer output menggunakan fungsi aktivasi Softmax untuk mengklasifikasikan ke empat kelas DML:

$$\hat{y} = \text{Softmax}(W_3 \times z_2 + b_3) \quad (6)$$

dengan  $\hat{y} \in \mathbb{R}^4$  merupakan distribusi probabilitas terhadap masing-masing kelas (SELECT, INSERT, UPDATE, DELETE). Bobot dan bias dari seluruh lapisan ( $W_1, W_2, W_3, b_1, b_2, b_3$ ) diperoleh selama training menggunakan backpropagation.

Fungsi loss yang digunakan adalah categorical cross-entropy dirumuskan sebagai:

$$L(y, \hat{y}) = -\sum_i (y_i \times \log(\hat{y}_i)), \quad i = 1 \dots 4 \quad (7)$$

dengan  $y$  adalah vektor label one-hot dari kelas sebenarnya, dan  $\hat{y}_i$  adalah probabilitas hasil prediksi model untuk kelas ke- $i$ . Proses optimisasi dilakukan menggunakan algoritma Adam dengan learning rate awal 0.001 dan batch size 32. Untuk mencegah overfitting, diterapkan dropout sebesar 0.5 di setiap lapisan dense, serta early stopping

berdasarkan akurasi pada data validasi.

Model usulan ditraining dengan sebuah dataset yang terdiri dari 1600 kalimat berbahasa Indonesia yang telah dianotasi sesuai jenis operasi DML-nya. Sebanyak 1280 data digunakan sebagai data latih, sedangkan 320 sisanya sebagai data uji. Evaluasi dilakukan menggunakan metrik akurasi, precision, recall, F1-score, dan confusion matrix untuk mengamati distribusi kesalahan antar kelas. Implementasi sistem dilakukan dengan framework TensorFlow/Keras.

### III. HASIL DAN PEMBAHASAN

#### A. Dataset dan Skenario Uji Coba

Dataset yang digunakan dalam penelitian ini terdiri dari 1.600 kalimat berbahasa Indonesia yang telah dianotasi sesuai dengan jenis operasi Data Manipulation Language (DML). Kalimat-kalimat ini dikumpulkan dari tiga domain, yaitu pendidikan, e-commerce, dan layanan publik. Domain ini dipilih untuk mencerminkan keragaman struktur bahasa. Setiap kalimat diberikan label dengan salah satu dari empat kelas DML, yaitu SELECT, INSERT, UPDATE, atau DELETE. Proses anotasi dilakukan secara manual.

Untuk meningkatkan reliabilitas hasil evaluasi, dilakukan teknik validasi silang (cross-validation) menggunakan skema 5-fold stratified cross-validation. Dataset dibagi ke dalam lima subset dengan distribusi label seimbang. Pada setiap fold, empat subset digunakan untuk pelatihan dan satu subset untuk pengujian, sehingga seluruh data berkesempatan menjadi data uji sebanyak satu kali. Hasil dari lima fold kemudian dirata-rata untuk memperoleh performa model secara keseluruhan. Pendekatan ini lebih stabil dibandingkan evaluasi satu kali split karena mengurangi ketergantungan pada pemilihan data uji tertentu.

Pelatihan dilakukan dengan maksimal selama 30 epoch dengan early stopping berdasarkan akurasi validasi. Proses eksperimen dijalankan menggunakan TensorFlow dengan akselerasi GPU untuk efisiensi waktu pelatihan. Parameter model seperti learning rate, batch size, dan dropout ditentukan melalui percobaan awal (manual tuning) pada data latih.

#### B. Hasil Evaluasi Pendekatan Usulan

Model BiLSTM dengan neural classifier diuji menggunakan skema 5-fold cross-validation dan dievaluasi menggunakan metrik akurasi, precision, recall, dan F1-score. Performa rata-rata dari kelima fold disajikan pada Tabel 1 berikut:

TABEL I  
HASIL RATA-RATA 5-FOLD CROSS-VALIDATION

Kelas DML	PRECISION	Recall	F1-score
SELECT	0,95	0,93	0,94
INSERT	0,91	0,92	0,91
UPDATE	0,88	0,86	0,87
DELETE	0,87	0,88	0,87
Rata-rata Makro	0,90	0,90	0,90

Sedangkan detail untuk masing-masing fold ditunjukkan pada tabel 2 berikut ini.

TABEL 2  
TABEL DETAIL K-FOLD

Fold	SELECT (F1)	INSERT (F1)	UPDATE (F1)	DELETE (F1)	Rata-rata F1
1	0,94	0,90	0,86	0,87	0,89
2	0,93	0,91	0,88	0,86	0,89
3	0,95	0,92	0,87	0,88	0,91
4	0,94	0,91	0,86	0,89	0,90
5	0,94	0,91	0,88	0,87	0,90
Akumulasi	0,94	0,91	0,87	0,87	0,90

Hasil evaluasi tiap fold menunjukkan konsistensi performa model dalam mengklasifikasikan kalimat ke dalam jenis operasi DML. Rata-rata F1-score dari lima fold adalah 0,90, dengan deviasi antar fold yang relatif kecil yaitu +/- 0.01. Kelas SELECT secara konsisten memberikan performa tertinggi, sementara kelas UPDATE dan DELETE cenderung memiliki nilai F1-score sedikit lebih rendah namun tetap stabil. Hal ini menunjukkan bahwa model mampu generalisasi dengan baik terhadap data uji yang berbeda-beda.

Akurasi keseluruhan dari lima fold mencapai 90,8%, dengan deviasi standar sekitar  $\pm 0.9$  antar fold. Hasil ini menunjukkan bahwa model memiliki performa yang stabil pada berbagai subset data. Nilai F1-score tertinggi terdapat pada kelas SELECT, yang menandakan bahwa struktur kalimat SELECT relatif lebih eksplisit dan konsisten. Sebaliknya, kelas UPDATE dan DELETE memiliki F1-score yang lebih rendah karena ekspresi linguistiknya yang lebih variatif dan sering kali tumpang tindih.

Analisis confusion matrix menunjukkan bahwa sebagian besar kesalahan klasifikasi terjadi antara kelas UPDATE dan DELETE. Hal ini cukup wajar karena dalam konteks bahasa Indonesia banyak kalimat yang bermakna penghapusan sekaligus bermakna pembaruan data. Meski demikian, proporsi kesalahan ini tidak signifikan dan masih berada dalam rentang toleransi performa klasifikasi.

### C. Pembahasan

Berdasarkan hasil evaluasi, pendekatan BiLSTM yang dikombinasikan dengan neural classifier terbukti efektif dalam mengklasifikasikan jenis operasi DML dari kalimat berbahasa Indonesia. BiLSTM berperan penting dalam mengkodekan konteks dua arah dalam kalimat. Sementara itu layer MLP mampu membedakan pola representatif antar kelas secara non-linear. Performa tinggi pada kelas SELECT dan INSERT menunjukkan bahwa sistem bekerja optimal pada struktur kalimat yang eksplisit dan umum digunakan.

Kalimat yang bermakna UPDATE dan DELETE menunjukkan adanya tantangan yang perlu diselesaikan untuk penelitian selanjutnya. Kalimat-kalimat ini sering kali menggunakan ekspresi yang mirip secara sintaktik namun berbeda secara semantik. Sebagai contoh, kalimat “Perbarui data pelanggan yang sudah tidak aktif” dan “Hapus data pelanggan yang sudah tidak aktif” memiliki pola kata yang serupa tetapi makna tindakan yang berbeda. Dalam hal ini, model yang hanya mengandalkan kata kunci eksplisit cenderung melakukan kesalahan.

Penggunaan skema k-fold dalam evaluasi juga memberikan gambaran yang lebih realistis mengenai generalisasi model. Rata-rata metrik dari lima fold menunjukkan stabilitas performa yang cukup baik dan minim overfitting. Hal ini menunjukkan bahwa model tidak hanya belajar dari pola-pola spesifik di data latih, tetapi mampu menangkap struktur linguistik yang lebih umum.

Sebagai tindak lanjut, pendekatan ini dapat ditingkatkan dengan mengganti word embedding konvensional menjadi embedding kontekstual seperti IndoBERT, yang mampu menangkap makna kata berdasarkan konteks kalimat. Selain itu, penambahan data dari domain lain dan penggunaan teknik ensemble juga berpotensi meningkatkan performa model secara keseluruhan.

## IV. KESIMPULAN

Pendekatan BiLSTM yang dikombinasikan dengan neural classifier terbukti efektif dalam mengklasifikasikan jenis operasi DML pada kalimat berbahasa Indonesia. Model ini mampu menangkap konteks dua arah dan membedakan pola antar kelas secara non-linear dengan performa optimal terutama pada kelas SELECT dan INSERT. Meski terdapat tantangan pada kelas UPDATE dan DELETE yang memiliki pola sintaktik mirip namun makna berbeda, kemampuan representasi kontekstual BiLSTM menjadi faktor penting dalam meningkatkan akurasi. Evaluasi menggunakan skema k-fold menunjukkan stabilitas performa yang baik dan kemampuan generalisasi yang minim risiko overfitting.

Untuk pengembangan selanjutnya, pendekatan selanjutnya disarankan menggunakan embedding kontekstual seperti IndoBERT agar pemahaman makna kata dalam konteks kalimat lebih baik dan akurasi klasifikasi meningkat. Selain itu, perluasan dataset dari berbagai domain serta penggunaan teknik ensemble perlu dieksplorasi agar dapat meningkatkan performa model secara keseluruhan.

### UCAPAN TERIMA KASIH

Ucapan terimakasih kami sampaikan kepada tim jurnal JUPI karena sudah memberikan kesempatan untuk submit di jurnal JUPI.

### DAFTAR PUSTAKA

- [1] R. Zhong, S. Su, and H. Chen, ‘SketchRefine: Hierarchical Text-to-SQL via Sketch and Refine’, *Transactions of the Association for Computational Linguistics*, vol. 11, pp. 210–225, 2023.
- [2] T. Yu, R. Zhang, A. Konstantinov, and others, ‘Spider: A Large-Scale Human-Labeled Dataset for Complex and Cross-Domain Semantic Parsing and Text-to-SQL Task’, in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2018, pp. 3911–3921.
- [3] P. Wang, V. Kishore, B. Keller, and others, ‘RAT-SQL: Relation-Aware Schema Encoding and Linking for Text-to-SQL Parsers’, in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020, pp. 7564–7576.
- [4] D. Guo, P. Pasupat, and T. Jaim, ‘Towards Generating Complex Text-to-SQL using Sub-query Encoding’, in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2019, pp. 725–734.
- [5] A. Mohammadjafari, A. Maida, and R. Gottumukkala, ‘From Natural Language to SQL: Review of LLM-based Text-to-SQL Systems’, *arXiv preprint arXiv:2410.01066*, 2024.

- [6] T. Mahmud and others, 'A rule based approach for nlp based query processing', in 2nd IEEE International Conference on Electrical Information and Communication Technologies (EICT), 2015, pp. 78–82.
- [7] I. Androutsopoulos, G. D. Ritchie, and P. Thanisch, 'Natural language interfaces to databases—An introduction', *Journal of Natural Language Engineering*, vol. 1, no. 1, pp. 29–81, 1995.
- [8] D. Patel, 'SyntaxSQLNet: Syntax-tree-based Decoding for Text-to-SQL', Medium blog, 2019.
- [9] D. Lee, J. Yoon, J. Song, and others, 'One-Shot Learning for Text-to-SQL Generation', arXiv preprint arXiv:1905.11499, 2019.
- [10] A. Elgohary and others, 'Interactive Text-to-SQL via Editable Step-by-Step Explanations', in EMNLP 2023, 2023, p.
- [11] R. Zhong and others, 'Addressing Limitations of Encoder–Decoder Based Approach to Text-to-SQL', in COLING 2022, 2022, p.
- [12] X. Xu, C. Liu, and D. Song, 'SQLNet: Generating Structured Queries From Natural Language Without Reinforcement Learning', arXiv preprint arXiv:1711.04436, 2017.
- [13] M. authors Unknown, 'Improving Text-to-SQL with a Hybrid Decoding Method', *Entropy*, vol. 25, no. 3, p. 513, 2023.
- [14] H. Sankhla, 'A Comprehensive Survey of LLM-Based Text-to-SQL', dev.to, 2024.
- [15] N. Rajkumar, R. Li, and D. Bahdanau, 'Evaluating the Text-to-SQL Capabilities of Large Language Models', arXiv preprint arXiv:2204.00498, 2022.
- [16] A. Mohammadjafari, A. Maida, and R. Gottumukkala, 'From Natural Language to SQL: Review of LLM-based Text-to-SQL Systems', arXiv preprint arXiv:2410.01066, 2024.
- [17] A. Gao and others, 'Text-to-SQL Empowered by Large Language Models: A Benchmark on Prompt Engineering', *PVLDB*, vol. 17, p. 1132, 2023.
- [18] T. Wu and others, 'Intent recognition model based on sequential information and sentence features', *Neurocomputing*, 2023.
- [19] A. S. Varghese and V. Mahalakshmi, 'Bidirectional LSTM joint model for intent classification and named entity recognition in natural language understanding', *Human-centric Computing and Information Sciences*, vol. 10, no. 1, pp. 1–17, 2020.
- [20] P. Gupta, 'Building a Text Classification model using BiLSTM'. 2020. [Online]. Available: <https://medium.com/analytics-vidhya/building-a-text-classification-model-using-bilstm-c0548ace26f2>
- [21] C. O. Bilah and T. B. Adji, 'Deteksi Intent pada Teks Bahasa Indonesia Menggunakan Bidirectional LSTM', Undergraduate Thesis, Universitas Gadjah Mada, 2023. [Online]. Available: <https://etd.repository.ugm.ac.id/penelitian/detail/228084>
- [22] A. Kurniawan, A. Abdiansah, and A. S. Utami, 'NL2SQL untuk Chatbot dengan Semantic Parsing menggunakan Metode Berbasis Aturan', Universitas Sriwijaya, Technical Report, 2022.
- [23] D. W. Chandra, 'Pembuatan aplikasi penerjemah Indonesian Query Language ke dalam bentuk SQL dengan metode non-deterministic finite automata', PhD Thesis, Universitas Kristen Petra, 2006.
- [24] I. Androutsopoulos, G. D. Ritchie, and P. Thanisch, 'Natural language interfaces to databases—An introduction', *Journal of Natural Language Engineering*, vol. 1, no. 1, pp. 29–81, 1995.
- [25] D. Patel, 'SyntaxSQLNet: Syntax-tree-based Decoding for Text-to-SQL', Medium blog, 2019.
- [26] D. Lee, J. Yoon, J. Song, and others, 'One-Shot Learning for Text-to-SQL Generation', arXiv preprint arXiv:1905.11499, 2019.
- [27] A. Elgohary and others, 'Interactive Text-to-SQL via Editable Step-by-Step Explanations', in Proceedings of EMNLP 2023, 2023.
- [28] V. Zhong, C. Xiong, and R. Socher, 'Seq2SQL: Generating Structured Queries from Natural Language using Reinforcement Learning', in arXiv preprint arXiv:1709.00103, 2017.
- [29] R. Zhong and others, 'Addressing Limitations of Encoder–Decoder Based Approach to Text-to-SQL', in Proceedings of COLING 2022, 2022.
- [30] X. Xu, C. Liu, and D. Song, 'SQLNet: Generating Structured Queries From Natural Language Without Reinforcement Learning', arXiv preprint arXiv:1711.04436, 2017.