

MULTI-WD: Alat Kelengkapan Multibahasa untuk Data Wikidata

Mohammad Yani¹, Lilyan Arhatia Agustine¹, Iryanto¹

¹ Program Studi Rekayasa Perangkat Lunak, Jurusan Teknik Informatika, Politeknik Negeri Indramayu, Indramayu, Jawa Barat, 45252, Indonesia

[Diserahkan: 4 Juni 2024, Direvisi: 19 Agustus 2024, Diterima: 1 Oktober 2024]
Penulis Korespondensi: Mohammad Yani (email: mohammad.yani@polindra.ac.id)

INTISARI — Wikidata adalah sebuah *knowledge graph* (KG) yang mengalami perkembangan sangat pesat. Hal ini disebabkan oleh dua hal. Pertama, data pada Wikidata dapat diakses dan diedit oleh siapa pun. Kedua, fitur multibahasa yang ditawarkan oleh Wikidata. Fitur multibahasa ini memungkinkan data (entitas) dalam Wikidata dapat diakses menggunakan berbagai bahasa di dunia. Namun, ketidaklengkapan informasi sebuah data Wikidata secara multibahasa masih menjadi isu penting dalam fitur ini. Sebagai contoh, deskripsi dari data “bada reuteuk” dengan identitas (ID) “Q100606305” hanya tersedia dalam bahasa Indonesia saja, yaitu “makanan khas di Indonesia”, tetapi tidak tersedia dalam bahasa lain. Dengan demikian, data tentang “bada reuteuk” tidak dapat dikenali dalam bahasa selain bahasa Indonesia. Untuk mengatasi permasalahan ketidaklengkapan data Wikidata secara multibahasa tersebut, diusulkan sebuah alat yang bernama MULTI-WD. Sistem ini dibangun menggunakan dua fitur utama, yaitu pemrofilan bahasa dan penerjemahan data. Pemrofilan data dilakukan menggunakan kueri SPARQL dengan memanfaatkan API Wikidata. Pemrofilan data ini berfungsi untuk menampilkan profil data Wikidata dari sisi kemultibahasaannya. Sementara itu, untuk penerjemahan data, digunakan pustaka penerjemah bernama Translated Labs. Translated Labs dipilih karena sifatnya yang terbuka dan gratis serta hasil terjemahannya yang cukup baik. Hasil terjemahan kemudian dapat disimpan ke Wikidata. Pengujian sistem dilakukan oleh lima responden dari komunitas Wikidata dengan menggunakan pendekatan *black box*. Hasil pengujian menunjukkan bahwa fungsi utama MULTI-WD, seperti pemilihan kategori, penampilan statistik data, penerjemahan dan pembaruan data, dapat berjalan 100% serta mencapai efisiensi penerjemahan data hingga 300%, dibandingkan dengan penerjemahan data secara manual atau langsung dari laman Wikidata.

KATA KUNCI — Pemrofilan Wikidata, Multibahasa Wikidata, Pemrofilan Multibahasa, Kelengkapan Data.

I. PENDAHULUAN

Dalam satu dekade terakhir, penggunaan *knowledge graph* (KG) dalam penelitian teknologi jejaring semantik (*semantic web*) sangat tinggi. KG adalah sebuah basis data sekunder yang berisi sekumpulan deskripsi dari entitas-entitas yang saling terkait oleh sebuah relasi [1], [2]. Entitas-entitas dalam KG merepresentasikan objek, situasi, peristiwa, dan konsep abstrak di dunia nyata. Ada beberapa KG besar saat ini. Salah satu di antaranya, yang paling populer, adalah Wikidata. Wikidata adalah sebuah KG yang bersifat terbuka dan dapat diakses serta diedit oleh manusia maupun mesin. Wikidata memiliki lebih dari seratus juta data (entitas) dan ribuan properti [3]. Wikidata merupakan salah satu KG besar yang ada saat ini [4].

Wikidata sangat menarik untuk digunakan dalam beberapa penelitian tentang sistem *knowledge graph question answering* (KGQA) karena dua alasan. Alasan pertama, data dalam Wikidata bersifat terbuka, semua orang dapat mengakses dan mengeditnya [5]-[7], sehingga data pada Wikidata dapat berkembang dengan sangat cepat. Kemudian, alasan kedua adalah Wikidata mendukung fitur multibahasa. Atas dasar kedua alasan tersebut, Wikidata banyak digunakan dalam penelitian tentang teknologi jejaring semantik khususnya sistem KGQA yang menerapkan pendekatan multibahasa [8], [9].

Akan tetapi, saat ini fitur multibahasa pada Wikidata masih memiliki kekurangan dari sisi kelengkapannya (*completeness*). Isu *completeness* yang dimaksud adalah kondisi ketika sebuah data dalam Wikidata hanya tersedia dalam suatu bahasa tertentu, tetapi tidak tersedia dalam bahasa lain. Sebagai contoh, deskripsi (*description*) dari data “bada reuteuk” dengan ID

“Q100606305” hanya tersedia dalam bahasa Indonesia, yaitu “makanan khas di Indonesia”, tetapi tidak tersedia dalam bahasa lain. Kasus seperti ini membuat informasi tentang data “bada reuteuk” tidak dikenali oleh bahasa lain. Informasi yang dimaksud adalah *label*, *description*, dan *also known as* dari sebuah data Wikidata.

Setidaknya ada dua faktor penyebab terjadinya ketidaklengkapan informasi dari data Wikidata secara multibahasa ini. Pertama, jumlah pengguna/penyunting data Wikidata antarbahasa berbeda-beda. Jumlah distribusi penyunting dari negara-negara di Asia lebih sedikit daripada di Eropa [10]. Akibatnya, informasi tentang data Wikidata lebih dominan dalam bahasa-bahasa negara Eropa. Kedua, terbatasnya waktu yang ada untuk melengkapi informasi sebuah data Wikidata dalam bahasa lain. Gambar 1 memberikan contoh bahwa data “bada reuteuk” hanya lengkap dalam bahasa Indonesia, tetapi tidak lengkap dalam bahasa lain. Dalam contoh sebagaimana pada Gambar 1, untuk data dalam bahasa Inggris hanya tersedia pada *label* saja, tidak pada *description* dan *also known as*. Begitu juga dengan bahasa lain yang kelengkapan datanya masih belum terpenuhi.

Dari Gambar 1 dapat dilihat bahwa informasi tentang data “bada reuteuk” hanya tersedia lengkap dalam bahasa Indonesia (*field Indonesian* di kolom *language*) saja, sedangkan informasi dalam bahasa lain masih belum lengkap. Beberapa informasi yang masih kosong pada kolom *label*, *description*, dan *also known as* inilah yang menjadi isu dalam penelitian ini.

Terdapat satu penelitian terkait kemultibahasaannya data Wikidata [11]. Penelitian tersebut melakukan pelengkapan deskripsi Wikidata menggunakan data dari Wikipedia. Akan

Bada reuteuk (Q100606305)

No description defined

▼ In more languages

Configure

Language	Label	Description	Also known as
English	Bada reuteu	No description defined	
Indonesian	Bada reuteu	variasi makanan khas Indone	bada reuteuk
Javanese	Bada reuteu	No description defined	
Sundanese	Bada reuteu	No description defined	

All entered languages

Gambar 1. Tangkapan layar contoh ketidaklengkapan data pada Wikidata secara multibahasa.

tetapi, pendekatan tersebut sangat bergantung pada kelengkapan informasi dari Wikipedia karena pendekatan yang digunakan adalah *point-to-point* dalam satu bahasa, bukan menerjemahkan data [11].

Selain itu, ada dua penelitian terkait kelengkapan data pada Wikidata, yaitu COOL-WD [12] dan ProWD [13]. Akan tetapi, keduanya hanya berfokus pada kelengkapan data Wikidata pada kasus nonmultibahasa. COOL-WD berfokus pada kelengkapan siklus hidup dari Wikidata, sedangkan ProWD mengusulkan alat yang dapat mengukur derajat kelengkapan Wikidata berdasarkan *class-facet-attribute* (CFA). Dari kedua penelitian tersebut belum didapatkan sebuah penelitian tentang cara mengatasi isu ketidaklengkapan data Wikidata secara multibahasa.

Atas dasar hal tersebut, penelitian ini mengusulkan sebuah alat yang dapat melakukan pemrofilan data menggunakan pendekatan kueri berbasis semantik dan melengkapi informasi data Wikidata secara multibahasa menggunakan pustaka mesin penerjemah. Pendekatan kueri berbasis semantik ini sangat umum digunakan untuk melakukan perolehan data dari sebuah KG. Namun, berdasarkan studi literatur, pendekatan ini belum pernah diimplementasikan untuk menyelesaikan isu ketidaklengkapan data Wikidata secara multibahasa.

Tujuan penelitian ini adalah menyelesaikan permasalahan isu ketidaklengkapan data Wikidata secara multibahasa dengan menggunakan pendekatan kueri berbasis semantik dan pustaka mesin penerjemah. Terdapat dua kontribusi dari penelitian ini. Pertama, penelitian ini diharapkan dapat memberikan kemudahan bagi pengguna untuk melengkapi data Wikidata dalam bahasa yang belum tersedia. Kedua, diharapkan penelitian ini dapat menyediakan *dataset* Wikidata yang lengkap secara multibahasa untuk kebutuhan penelitian tentang sistem KGQA.

Selain itu, penelitian ini diharapkan dapat digunakan untuk melakukan pemrofilan data Wikidata untuk bahasa-bahasa dengan sumber daya rendah. Bahasa-bahasa dengan sumber daya rendah adalah data yang ketersediaannya dalam bahasa tertentu lebih sedikit dibandingkan dengan bahasa lainnya [14].

II. KEMULTIBAHASAAN DATA WIKIDATA

Terdapat satu penelitian terkait kemultibahasaan data Wikidata. Penautan *event* (*event linking*) deskripsi data Wikidata untuk semua bahasa menggunakan Wikipedia telah dilakukan [11]. Pada penelitian ini, deskripsi Wikidata diisi dengan hasil pencarian *event* sejenis yang terdapat pada Wikipedia. Kekurangan dari pendekatan ini adalah tidak semua data yang ada pada Wikidata tersedia pada Wikipedia. Selain itu, penelitian ini hanya mencari deskripsi dalam bahasa

yang sama, bukan menerjemahkan data dari satu bahasa ke bahasa lain.

Sementara itu, penelitian terkait alat kelengkapan data Wikidata ini adalah COOL-WD [12] dan ProWD [13]. COOL-WD adalah sebuah alat anotasi untuk memeriksa kelengkapan siklus hidup data Wikidata, sedangkan ProWD menyediakan sebuah *dashboard* yang melakukan pemrofilan kelengkapan pernyataan dari sebuah entitas dalam Wikidata menggunakan pendekatan CFA. Akan tetapi, kedua penelitian tersebut tidak melakukan pelengkapan data Wikidata secara spesifik pada fitur multibahasa. Kedua penelitian tersebut lebih berfokus pada kelengkapan subjek dan predikat sebuah data sebagai syarat minimal untuk dianggap sebagai data yang lengkap di dalam pernyataannya (*statement*). Untuk itu, pada penelitian ini diusulkan sebuah alat yang dapat digunakan untuk melakukan pemrofilan data Wikidata menggunakan pendekatan kueri berbasis semantik dan menerjemahkan informasi dari data Wikidata secara multibahasa menggunakan pustaka mesin penerjemah. Kontribusi utama penelitian ini adalah menyediakan data Wikidata yang lengkap secara multibahasa, sehingga dapat digunakan oleh peneliti sistem KGQA yang menerapkan konsep multibahasa. Sistem KGQA adalah sebuah sistem yang digunakan untuk mengakses data dari KG dengan menggunakan masukan pertanyaan dalam bahasa alami [15].

III. MULTI-WD

Pada bagian ini, arsitektur dan implementasi MULTI-WD dijelaskan. Pada bagian ini juga diberikan potongan-potongan program utama dari MULTI-WD ini.

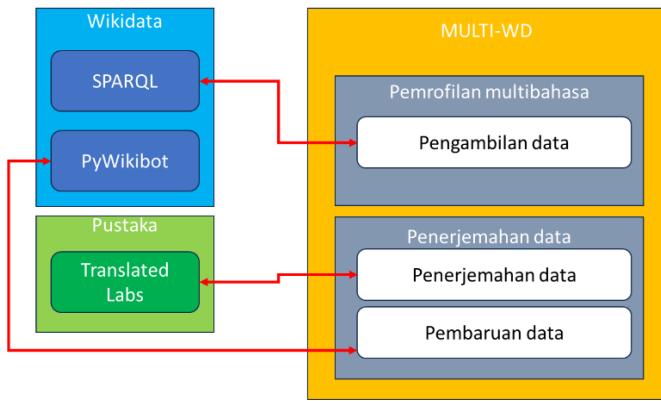
A. ARSITEKTUR MULTI-WD

Arsitektur MULTI-WD dirancang untuk tujuan melakukan pemrofilan data Wikidata menggunakan pendekatan kueri berbasis semantik dan penerjemahan informasi dari data Wikidata menggunakan pustaka mesin penerjemah. Terdapat dua fitur utama dalam MULTI-WD ini, yaitu pemrofilan multibahasa data Wikidata dan penerjemahan data Wikidata. Untuk mengimplementasikan kedua fitur tersebut, MULTI-WD dibangun dari tiga bagian utama, yaitu SPARQL, Translated Labs, dan Pywikibot. Rancangan arsitektur MULTI-WD ditunjukkan pada Gambar 2. SPARQL digunakan untuk mengambil data dari Wikidata, Translated Labs digunakan untuk menerjemahkan data dari bahasa asal ke bahasa tujuan, dan Pywikibot digunakan untuk mengakses serta menyunting data Wikidata melalui Python.

1) PEMROFILAN MULTIBAHASA

Fitur ini melakukan pemrofilan multibahasa yang meliputi informasi tentang *label*, *description*, dan *also known as* dari sebuah data Wikidata. Untuk kueri berbasis semantiknya, MULTI-WD menggunakan SPARQL. Kueri berbasis semantik adalah kueri yang digunakan untuk mengambil data dalam bentuk grafik (*graph*) yang tersedia pada sebuah KG. Grafik pada KG ini biasanya dalam bentuk *triple* (subjek, predikat, dan objek).

SPARQL digunakan oleh MULTI-WD untuk melakukan pemrofilan data Wikidata secara multibahasa. SPARQL adalah bahasa kueri yang digunakan untuk melakukan pengambilan data dalam bentuk *triple* dari sebuah KG [16]. SPARQL dapat digunakan untuk menjawab pertanyaan yang jawabannya diambil dari sebuah KG, baik dalam format *resource description framework* (RDF) maupun Web Ontology Language (OWL) [17]. SPARQL juga dapat digunakan untuk



Gambar 2. Arsitektur MULTI-WD.

melakukan kueri ke sebuah KG melalui *application programming interface* (API), sehingga data dapat diolah lebih lanjut [18]. SPARQL banyak digunakan sebagai jembatan berkomunikasi antara pengguna dengan KG dalam sebuah sistem KGQA [19].

Pada penelitian ini, data Wikidata yang diambil adalah data dalam beberapa bahasa resmi pada Perserikatan Bangsa-Bangsa (PBB), seperti bahasa Inggris, Perancis, Spanyol, serta bahasa Indonesia sebagai bahasa pengantar resmi pada UNESCO [20]. Keempat bahasa tersebut dipilih karena merupakan bahasa pengantar resmi dunia, baik di PBB maupun UNESCO, dan memiliki jumlah kontributor data Wikidata terbesar. Gambar 3 menunjukkan contoh pengambilan data (*ID* dan *label*) semua jenis makanan khas Indonesia dalam bahasa Indonesia melalui *endpoint* Wikidata. Untuk dapat menggunakan SPARQL melalui Python, MULTI-WD menggunakan SPARQLWrapper.

Pada Gambar 3 terlihat bahwa perintah kueri terbagi menjadi dua bagian: klausul *SELECT* mengidentifikasi variabel-variabel yang akan muncul dalam hasil kueri; dan klausul *WHERE* menyediakan pola grafik dasar yang akan dicocokkan dengan grafik (*graph*) yang ada dalam KG. *Syntax* sebagaimana pada Gambar 3 dapat dimodifikasi untuk mendapatkan data dalam bahasa lain dengan mengubah pengaturan bahasanya pada pola grafik *SERVICE wikibase:label*.

Dalam contoh pada Gambar 3, kueri akan memunculkan dua nilai yang akan ditampilkan melalui variabel *?makanan* dan *?makananLabel* yang diambil dari dua pola grafik. Pola grafik dipresentasikan dalam bentuk *triple*. *Triple* adalah sebuah format berbentuk (subjek, predikat, objek). Dua pola grafik ditunjukkan pada baris ke-3 dan ke-4 pada Gambar 3.

Pola grafik pertama seperti pada baris ke-3 dalam Gambar 3 merujuk pada semua entitas yang merupakan “*instant* dari” (P31) entitas “*food*” (Q2095), sedangkan pola grafik kedua sebagaimana pada baris ke-4 dalam Gambar 3 merujuk pada entitas “berjenis” “*food*” yang berasal dari “*country*” (P17) “Indonesia” (Q252). *ID* yang diawali huruf “P” menunjukkan properti, sedangkan huruf “Q” menunjukkan entitas.

Pola grafik *SERVICE wikibase:label* { *bd:serviceParam wikibase:language "id"*. } adalah pola grafik yang digunakan untuk menyaring informasi dalam bahasa Indonesia (“id”) saja. Perubahan bahasa yang digunakan dapat dilakukan dengan cara mengubah inisial bahasa “id” ke inisial bahasa lain, misalnya “en” untuk bahasa Inggris. Hasil dari kueri tersebut kemudian ditampilkan ke variabel *?makanan* dan *?makananLabel*, seperti diperlihatkan pada Gambar 4.

```
Wikidata Query Service Examples Help More tools
1 SELECT DISTINCT ?makanan ?makananLabel WHERE {
2   SERVICE wikibase:label { bd:serviceParam wikibase:language "id" . }
3   ?makanan (wdt:P31/(wdt:P279*)) wd:Q2095.
4   ?makanan wdt:P17 wd:Q252.
5 }
```

Gambar 3. Kueri SPARQL untuk mengambil semua data berjenis makanan khas Indonesia dalam bahasa Indonesia.

Hasil

Tampilkan: 10 Cari:

Butir	Label	Aksi
Q100281422	Megono	Edit
Q100701181	Beras rendang	Edit
Q104621347	Asam padeh ikan	Edit
Q104629902	Karabu Baluik	Edit
Q104641214	Kue sagun	Edit
Q108096758	Q108096758	Edit
Q10954885	dodol Betawi	Edit
Q10978475	Pakasam	Edit
Q10986174	Sate Madura	Edit
Q11017447	Bandeng presto	Edit

Sebelumnya 1 2 3 4 5 ... 113 Selanjutnya

Gambar 4. Tangkapan layar hasil kueri data makanan Indonesia berbahasa Indonesia.

2) PENERJEMAHAN DATA

Fitur ini memiliki dua subfitur, yaitu penerjemahan data dan pembaruan data. Untuk fitur penerjemahan data, MULTI-WD menggunakan pustaka mesin penerjemah Translated Labs. Translated Labs dipilih karena pustaka ini bersifat sumber terbuka (*open source*), sehingga setiap orang dapat mengaksesnya dari mana pun. Translated Labs adalah sebuah alat berbasis *artificial intelligence* (AI) yang dapat digunakan untuk menerjemahkan satu bahasa ke bahasa lain yang dapat diakses melalui API. Translated Labs ini menggunakan model Modern Machine Translation (MMT) yang diproduksi oleh MMT pada tahun 2018 [21]. Pada fitur ini, informasi mengenai *label*, *description*, dan *also known as* dari data Wikidata asal akan digunakan sebagai masukan dari Translated Labs yang kemudian diterjemahkan menggunakan MMT ke dalam bahasa tujuan. Hasil terjemahan yang diperoleh dari fitur penerjemahan data ini tidak secara langsung disimpan ke dalam Wikidata. Untuk memperbarui informasi dari data yang diterjemahkan, MULTI-WD menerapkan fitur pembaruan data. Fitur ini melakukan pembaruan data untuk informasi-informasi yang diterjemahkan ke dalam Wikidata. Untuk melakukan proses tersebut, MULTI-WD menggunakan PyWikibot. PyWikibot adalah salah satu pustaka Python yang dapat digunakan untuk menghubungkan Python dengan data yang ada di Wikidata. Proses MULTI-WD diperlihatkan pada Gambar 5.

B. IMPLEMENTASI MULTI-WD

Pada bagian ini dijelaskan tentang implementasi MULTI-WD yang meliputi implementasi sistem dan *use case* pada MULTI-WD. MULTI-WD dijalankan pada komputer 64 bit dengan 11th Gen Intel® Core™ i7-1165G7 @ 2,80 GHz dan memori RAM 16 GB. MULTI-WD menggunakan Python sebagai *back-end* serta SPARQL sebagai layanan kueri untuk mengakses data Wikidata melalui sebuah pustaka PyWikibot.



Gambar 5. Alur proses pemrofilan entitas Wikidata MULTI-WD.

Sementara itu, untuk *front-end*-nya, MULTI-WD menggunakan *framework* Laravel 8.

1) IMPLEMENTASI SISTEM

MULTI-WD dibangun dalam bentuk aplikasi berbasis web untuk melakukan pemrofilan data dan penerjemahan informasi data Wikidata. Alat ini melakukan pemrofilan data menggunakan kueri SPARQL. Di sisi lain, untuk penerjemahan informasi dari data Wikidata, MULTI-WD menggunakan pustaka mesin penerjemah Translated Labs. PyWikibot digunakan untuk menghubungkan Python dan Wikidata.

2) USE CASE PADA MULTI-WD

Di sini dijelaskan beberapa *use case* yang dapat dilakukan oleh MULTI-WD. Pada bagian ini pula dijelaskan cara MULTI-WD dapat menyajikan data dari Wikidata, memilih jenis data yang akan diambil, memonitor statistik data berdasarkan bahasanya, menerjemahkan data, dan memperbaruinya ke dalam Wikidata.

Use case pertama adalah pengambilan data dari Wikidata. Pengambilan data Wikidata dari Python dilakukan menggunakan SPARQL. Data diambil berdasarkan kriteria yang diberikan dalam kueri. Contoh kueri untuk pengambilan data dapat dilihat pada Gambar 3.

Use case kedua adalah pemilihan jenis data yang akan diambil dari Wikidata. Pada *use case* ini, pengguna dapat mengambil data dari Wikidata berdasarkan jenis dari datanya, misalnya tarian, makanan, lagu, dan rumah adat. Pemilihan jenis data ini akan menampilkan seluruh data dengan jenis tersebut. Sebagai contoh, Gambar 6 menampilkan data yang berjenis rumah adat.

Pada klausul *SELECT*, kueri akan mengambil dua data yang akan disimpan ke dalam variabel `?rumah` dan `?rumahLabel`. Kemudian, pada klausul *WHERE*, kueri akan mencocokkan beberapa pola grafik. Pada kueri ini, pola grafik yang akan dicocokkan adalah empat pola grafik. Keempat pola grafik tersebut sebagaimana pada baris ke-3, ke-5, ke-7, dan ke-8 di Gambar 6. Secara umum, pola sebagaimana pada baris-baris tersebut terdiri atas sebuah *triple*. *Triple* `?rumah wdt:P31 wd:Q3947` menunjukkan bahwa `?rumah` adalah subjek, `wdt:P31` adalah predikat, dan `wd:Q3947` adalah objek. Pada *triple* tersebut, terdapat dua *PREFIX*, yaitu `"wdt"` dan `"wd"`. *PREFIX* `"wdt"` digunakan untuk menunjukkan properti Wikidata, sedangkan *PREFIX* `"wd"` untuk entitas Wikidata.

Pola grafik sebagaimana pada baris ke-3 di Gambar 6 menunjukkan bahwa kueri akan mencocokkan semua data pada Wikidata yang memiliki *triple* `?rumah wdt:P31 wd:Q3947`. Properti `"P31"` merepresentasikan sebuah properti Wikidata *"instant dari"*, sedangkan entitas `"Q3947"` merepresentasikan sebuah entitas Wikidata *"house"*. Dengan pola grafik ini, semua data yang merupakan *"instant dari"* entitas *"house"* akan disimpan ke dalam variabel `?rumah`.

Pola grafik sebagaimana pada baris ke-5 di Gambar 6 akan mengambil semua data yang merupakan `"P31"` (*"instant dari"*)

```

1 SELECT ?rumah ?rumahLabel WHERE {
2   SERVICE wikibase:label { bd:serviceParam wikibase:language "[AUTO_LANGUAGE], id". }
3   {?rumah wdt:P31 wd:Q3947}
4   UNION
5   {?rumah wdt:P31 wd:Q7419654}
6   UNION
7   {?rumah wdt:P31 wd:Q811979}
8   ?rumah wdt:P17 wd:Q252
9 }
    
```

Gambar 6. Kueri SPARQL untuk menampilkan data berjenis rumah adat.

entitas `"Q7419654"` (*"rumah adat"*) yang akan disimpan ke dalam variabel `?rumah`. Pola grafik sebagaimana pada baris ke-7 di Gambar 6 mengambil semua data yang merupakan `"P31"` (*"instant dari"*) entitas `"Q811979"` (*"architectural structure"*) yang kemudian disimpan dalam variabel `?rumah`. Pola grafik sebagaimana pada baris ke-7 di Gambar 6 mengambil semua data yang merupakan `"P17"` (*"country"*) entitas `"Q252"` (*"Indonesia"*) yang disimpan juga dalam variabel `?rumah`.

Pola grafik sebagaimana pada baris ke-3 sampai ke-5 dalam Gambar 6 masing-masing menghasilkan satu himpunan hasil kueri yang kemudian digabungkan dengan menggunakan *syntax UNION*. Untuk hasil *UNION* tersebut, semuanya harus memenuhi pola grafik sebagaimana pada baris ke-7 di Gambar 6, yaitu semua entitas yang berasal dari negara Indonesia (`wdt:P17 wd:Q252`).

Syntax SERVICE `wikibase:label { bd:serviceParam wikibase:language "[AUTO_LANGUAGE],id". }` merujuk pada semua data yang ditulis dalam bahasa Indonesia. Jika ingin diambil data yang ditulis dalam bahasa Inggris, `"[AUTO_LANGUAGE],id"` dapat diubah menjadi `"[AUTO_LANGUAGE],en"`. Dua karakter terakhir `"id"` dan `"en"` merepresentasikan bahasa yang digunakan untuk Indonesia dan bahasa Inggris. Selanjutnya, hasil dari *syntax SERVICE* ini disimpan dan ditampilkan pada variabel `?rumahLabel`.

Use case ketiga adalah penyajian statistik jumlah data berdasarkan jenis dan bahasanya. Data statistik diperoleh dengan cara menjumlahkan data berdasarkan bahasanya dan ditampilkan dalam bentuk grafik batang. Data statistik dibagi berdasarkan informasi *label*, *description*, dan *also known as*.

Use case keempat adalah penerjemahan informasi dari data ke bahasa tujuan. MULTI-WD melakukan penerjemahan informasi *label*, *description*, dan *also known as* dari bahasa asal ke bahasa tujuan menggunakan pustaka Translated Labs.

Use case kelima adalah memperbarui data Wikidata yang telah diterjemahkan oleh *use case* keempat. Untuk melakukan pembaruan/penyuntingan data pada Wikidata, MULTI-WD menggunakan pustaka PyWikibot.

IV. DEMONSTRASI SISTEM

Pada bagian ini, demonstrasi sistem dilakukan, mulai dari skenario demonstrasi, contoh *running*, pengujian, dan validasi hasil demonstrasi. Kode lengkap MULTI-WD dapat diakses di <https://github.com/moh-yani/multi-wd/>.

A. SKENARIO DEMONSTRASI

Demonstrasi dilakukan pada *use case-use case* yang ada pada MULTI-WD, meliputi analisis statistik data Wikidata, penerjemahan data, dan pembaruan data. Subbagian berikut lebih memperjelas demonstrasi.

B. CONTOH RUNNING

Pada bagian ini, contoh *running* yang akan digunakan adalah contoh sebagaimana Gambar 3, yaitu data tentang makanan khas Indonesia. Tampilan pertama dari MULTI-WD

adalah memilih jenis data yang akan diprofil. Misalnya, dalam contoh ini adalah makanan khas. Setelah memilih jenis data makanan khas, MULTI-WD akan menampilkan informasi tentang data dan statistiknya untuk jenis data makanan khas tersebut.

Setelah proses pemilihan jenis data, MULTI-WD menampilkan statistik data Wikidata dalam empat bahasa yang diklasifikasikan berdasarkan *label*, *description*, dan *also known as*. Gambar 7 hanya menunjukkan salah satu contoh data statistik berdasarkan deskripsi untuk empat bahasa saja.

Dari Gambar 7 diketahui bahwa deskripsi data “makanan khas” dalam bahasa Spanyol memiliki tingkat kelengkapan yang paling rendah dari ketiga bahasa lainnya, diikuti oleh bahasa Perancis dan bahasa Inggris. Bahasa Spanyol dan bahasa Perancis yang memiliki sedikit ketersediaan data ini disebut sebagai bahasa dengan sumber daya rendah dalam konteks KG. Dari informasi tentang data yang dihasilkan sebagaimana pada Gambar 4, dapat dilakukan penyuntingan bahasa melalui aksi “edit”. Penyuntingan bahasa yang dimaksud adalah penerjemahan data dari bahasa asal ke bahasa tujuan melalui tombol “Translate Text”. Setelah penyuntingan selesai, MULTI-WD melalui tombol “Edit Data” dapat memperbarui data Wikidata secara *realtime*. Proses penyuntingan dan penerjemahan data ditunjukkan pada Gambar 8. Hasil data “Bada reuteuk” pada Gambar 1 yang telah diterjemahkan melalui Gambar 8 dapat dilihat pada <https://www.wikidata.org/wiki/Q100606305>.

C. PENGUJIAN

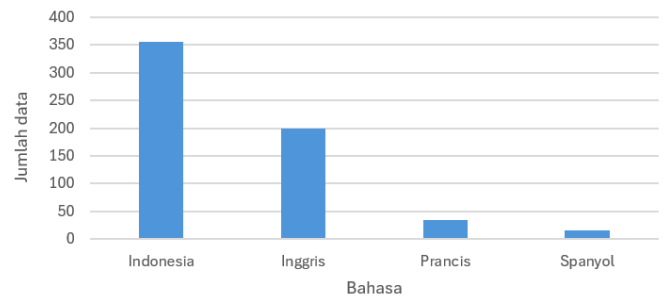
Pada tahap ini, pengujian dilakukan oleh lima anggota komunitas Wikidata Indonesia yang sekaligus merupakan pengguna/penyunting data Wikidata. Wikidata Indonesia adalah organisasi nirlaba dan mitra lokal dari Yayasan Wikimedia untuk mengelola proyek-proyek Wikipedia dan wiki-wiki lainnya. Pada tahap ini, pengujian yang dilakukan adalah pengujian fungsional dari *use case* yang ada pada MULTI-WD menggunakan *black box*. Pengujian ini bertujuan untuk mengetahui bahwa sistem dapat melakukan pemprofilan (penampilan statistik) data, penerjemahan data, dan pembaruan data Wikidata secara *realtime* atau tidak.

Pengujian ini dilakukan pada empat jenis *instant* Wikidata, yaitu makanan khas, tarian tradisional, senjata tradisional, dan rumah adat. Dari masing-masing jenis *instant* diambil sepuluh *instant* secara acak untuk dilakukan pemprofilan (penampilan statistik), penerjemahan, dan pembaruan data secara *realtime*. Masing-masing pengujian diberi kebebasan dalam menentukan secara acak *instant-instant* yang akan diuji berdasarkan jenis *instant*-nya. Pengujian diberi kebebasan dalam menentukan sumber dan tujuan bahasa dalam proses penerjemahan sebuah *instant* serta diberi kebebasan untuk menyimpan hasil penerjemahan ke dalam Wikidata atau tidak. Pada bagian di bawah ini disampaikan beberapa kasus uji (*test case*) pengujian MULTI-WD yang meliputi tujuan, hasil yang diharapkan, serta hasil pengujian untuk setiap kasus uji.

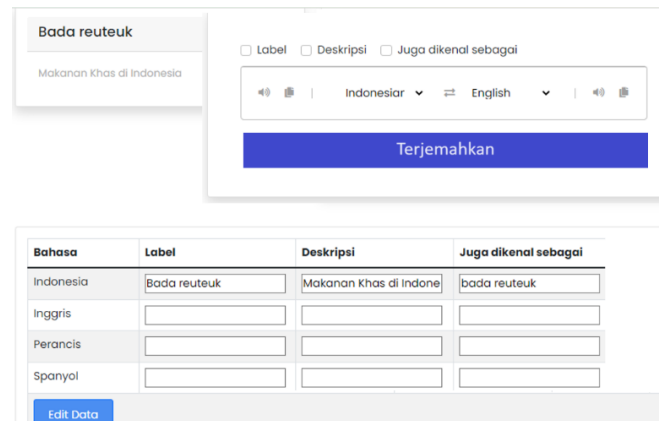
1) PENGUJIAN PEMILIHAN KATEGORI

Tujuan dari pengujian ini adalah untuk menguji fungsi dari MULTI-WD, dapat melakukan pemilihan kategori data dan menampilkannya sesuai dengan kategori tersebut atau tidak. Hasil yang diharapkan dari pengujian ini adalah MULTI-WD dapat menampilkan data sesuai kategori yang dipilih, yang meliputi *label*, *description*, dan *also known as*. Hasil pengujian pada Tabel I menunjukkan bahwa proses pengambilan data

Statistik Deskripsi Tentang Makanan Khas Indonesia



Gambar 7. Tampilan statistik deskripsi data “makanan khas” dalam empat bahasa.



Gambar 8. Tampilan halaman penerjemahan data dan pembaruan data.

berdasarkan kategori berhasil dilakukan. Skenario pengujian pemilihan kategori ini ditunjukkan pada Tabel I.

2) PENGUJIAN PENAMPILAN STATISTIK MULTIBAHASA DARI ENTITAS WIKIDATA

Tujuan pengujian ini adalah untuk mengetahui MULTI-WD dapat menampilkan statistik data Wikidata berdasarkan jenis data yang dipilih untuk empat bahasa dalam bentuk grafik batang atau tidak. Hasil yang diharapkan dari pengujian ini adalah MULTI-WD dapat menampilkan statistik data dalam bentuk grafik batang. Hasil pengujian sebagaimana pada Tabel II menunjukkan bahwa fungsi ini dapat berjalan sesuai dengan yang diharapkan, yaitu menampilkan statistik data Wikidata dalam grafik batang.

3) PENGUJIAN PENERJEMAHAN DATA ENTITAS

Tujuan dari pengujian ini adalah untuk mengetahui kemampuan MULTI-WD menerjemahkan data dari bahasa asal ke bahasa tujuan; serta menguji kesesuaian hasil terjemahan dengan kaidah penyuntingan data Wikidata. Hasil pengujian sebagaimana pada Tabel III menunjukkan bahwa sistem dapat menerjemahkan data dari bahasa asal ke bahasa tujuan dan hasil terjemahan telah sesuai dengan kaidah penyuntingan data pada Wikidata. Kaidah penyuntingan dapat dilihat di <https://github.com/moh-yani/multi-wd/>. Skenario pengujian penerjemahan data ini dapat dilihat pada Tabel III.

4) PENGUJIAN PEMBARUAN DATA KE WIKIDATA

Tujuan dari pengujian ini adalah menguji kemampuan sistem dalam melakukan pembaruan data Wikidata secara *realtime* dan memastikan tidak terjadi *bug*. Hasil yang diharapkan dari pengujian ini adalah sistem dapat melakukan pembaruan data Wikidata secara *realtime* dan tanpa *bug*. Hasil pengujian sebagaimana data yang dapat dilihat pada

TABEL I
SKENARIO KASUS UJI PEMILIHAN KATEGORI

Identifikasi		Kasus Uji		
Kasus uji		Pemilihan kategori		
Tujuan		Memeriksa tampilan setelah memilih kategori (menampilkan hasil data kategori dan menampilkan kelengkapan data <i>label</i> , <i>description</i> , dan juga <i>also known as</i> , berdasarkan bahasa)		
Skenario				
1. Klik “Pilih Opsi” untuk memilih salah satu kategori budaya 2. Menampilkan hasil pilih salah satu kategori budaya 3. Menampilkan hasil kelengkapan data dalam bentuk diagram				
Hasil yang Diharapkan				
1. Menampilkan data entitas sesuai kategori yang dipilih 2. Menampilkan diagram yang menyajikan data statistik sebuah entitas				
Statistik				
Jumlah Responden	Jumlah Data Uji Kategori per Responden	Sukses	Gagal	Keterangan
5	4	20	0	Sesuai tujuan pengujian

TABEL II
SKENARIO KASUS UJI PENAMPILAN STATISTIK MULTIBAHASA DARI ENTITAS WIKIDATA

Identifikasi		Kasus Uji		
Kasus uji		Penampilan statistik multibahasa dari entitas Wikidata		
Tujuan		Memeriksa tampilan setelah pengguna memilih salah satu entitas (sistem akan menampilkan statistik datanya secara multibahasa)		
Skenario				
1. Klik “Tampilkan Daftar Entitas” pada menu “Pilih Opsi” 2. Klik “Nama Entitas” pada baris yang berisi daftar entitas				
Hasil yang Diharapkan				
Sistem menampilkan statistika data kelengkapan multibahasa sebuah entitas dalam bentuk diagram				
Statistik				
Jumlah Responden	Jumlah Data Uji Entitas per Responden	Sukses	Gagal	Keterangan
5	40	200	0	Sesuai tujuan pengujian

<https://www.wikidata.org/wiki/Q100606305> menunjukkan bahwa sistem dapat berjalan sesuai dengan fungsinya, yaitu sebagai penyunting bahasa pada data Wikidata secara *realtime* dan tanpa *bug*. Skenario pengujian penerjemahan data ini disajikan pada Tabel IV.

Proses pembaruan data entitas Wikidata ke laman Wikidata secara umum dapat dilakukan. Namun, keberhasilan dari pembaruan ini sangat bergantung pada kondisi API Wikidata yang ada. Pembaruan data bisa saja gagal dilakukan jika API dalam kondisi *time out*. Hasil pengujian pembaruan data ke laman Wikidata sebagaimana pada Tabel IV adalah ketika kondisi API tidak dalam keadaan *time out*.

D. VALIDASI

Validasi dilakukan oleh lima anggota komunitas Wikidata Indonesia. Validasi dilakukan menggunakan pendekatan pengamatan secara langsung dan pengecekan hasil

TABEL III
SKENARIO KASUS UJI PENERJEMAHAN DATA ENTITAS

Identifikasi		Kasus Uji		
Kasus uji		Penerjemahan data entitas		
Tujuan		Memeriksa apakah apakah sistem dapat menerjemahkan data sebuah entitas dari bahasa asal ke bahasa tujuan dan meletakkannya (hasil terjemahan) ke dalam kolom edit		
Skenario				
1. Pilih salah satu entitas yang ingin diterjemahkan 2. Pilih bahasa asal dan tujuan pada menu 3. Klik menu “Terjemahkan”				
Hasil yang diharapkan				
Sistem dapat menerjemahkan data entitas dari bahasa asal ke bahasa tujuan dan meletakkannya di kolom edit				
Statistik				
Jumlah Responden	Jumlah Data Uji Entitas per Responden	Sukses	Gagal	Keterangan
5	40	200	0	Sesuai tujuan pengujian

TABEL IV
SKENARIO KASUS UJI PEMBARUAN DATA KE WIKIDATA

Identifikasi		Kasus Uji		
Kasus uji		Pembaruan data entitas ke Wikidata		
Tujuan		Memeriksa kemampuan sistem untuk benar-benar memperbarui data entitas di laman Wikidata		
Skenario				
1. Klik tombol “Edit Data” pada sebagaimana tampilan pada Gambar 8 2. Cek data entitas yang telah diperbarui di laman Wikidata				
Hasil yang Diharapkan				
Sistem dapat memperbarui data Wikidata di laman Wikidata				
Statistik				
Jumlah Responden	Jumlah Data Uji Entitas per Responden	Sukses	Gagal	Keterangan
5	40	200	0	Sesuai tujuan pengujian

penyuntingan data menggunakan MULTI-WD. Tujuan dari tahap ini adalah untuk memvalidasi sistem, sudah bekerja sesuai dengan fungsinya atau belum. Pemilihan metode validasi secara manual oleh manusia ini bertujuan untuk dapat memberikan hasil yang mendekati *gold standard*.

Validasi untuk *use case* pemilihan data dan penampilan data dilakukan dengan cara perbandingan melalui pengambilan data secara langsung menggunakan *endpoint* Wikidata di <https://query.wikidata.org/>. Hasil dikatakan valid jika dan hanya jika data yang dihasilkan oleh MULTI-WD ekuivalen dengan data yang dihasilkan dari *endpoint* Wikidata secara langsung. Statistik hasil validasi pada *use case* pemilihan dan penampilan data ditunjukkan pada Tabel I.

Validasi untuk *use case* penerjemahan data dilakukan dengan pengamatan secara langsung hasil terjemahan, sudah tertulis dalam format yang sesuai dengan kaidah penyuntingan data Wikidata atau belum. Kaidah penyuntingan dapat dilihat di <https://github.com/moh-yani/multi-wd/>. Hasil dikatakan valid jika dan hanya jika hasil terjemahan telah sesuai dengan kaidah penyuntingan data Wikidata. Statistik hasil validasi pada *use case* penerjemahan data dapat dilihat pada Tabel III.

Validasi untuk *use case* pembaruan data dilakukan dengan pengecekan secara langsung ke laman data Wikidata yang dimaksud. Misalnya, pengecekan hasil pembaruan data pada data “Bada reuteuk” dapat dilakukan melalui tautan <https://www.wikidata.org/wiki/Q100606305>. Hasil dikatakan valid jika dan hanya jika data yang telah diperbarui di laman Wikidata sama dengan data yang dihasilkan oleh MULTI-WD. Statistik hasil validasi pada *use case* pembaruan data dapat dilihat pada Tabel IV.

E. DISKUSI

Dari hasil pengujian terhadap beberapa kasus uji sebagaimana tersaji pada Tabel I sampai Tabel IV, secara umum, fungsi dari MULTI-WD, yang meliputi pengujian unit dan pengujian sistem, dapat berjalan sesuai dengan ekspektasi. Akan tetapi, secara khusus, pada fitur penerjemahan dan pembaruan data ke Wikidata ada beberapa catatan. Pertama, secara *default*, fitur penerjemahan data menghasilkan hasil terjemahan mesin, sehingga mungkin saja hasil terjemahan tidak berupa bahasa alami yang baik (*gold standard*) dan memerlukan proses penyuntingan kembali oleh pengguna. Oleh karena itu, pada proses ini MULTI-WD menyediakan fitur edit untuk menyunting data hasil terjemahannya.

Kedua, fitur pembaruan data ke Wikidata bisa saja tertahan sementara karena melampaui batas akses API ke Wikidata. Faktor eksternal ini tidak dapat dikendalikan oleh MULTI-WD. Hal yang dapat dilakukan adalah mengelola penanganan kesalahan yang terjadi agar pengguna tidak merasa telah terjadi kesalahan fatal yang dilakukan oleh MULTI-WD dengan cara memberikan pesan kepada pengguna tentang kendala yang terjadi saat melakukan pembaruan data ke Wikidata.

Meskipun demikian, dari sisi efisiensi MULTI-WD lebih efisien 300% jika dibandingkan dengan penerjemahan manual melalui laman Wikidata. MULTI-WD dapat menerjemahkan secara langsung data *label*, *description*, dan *also known as* secara langsung dalam satu kali proses, berbeda dengan penerjemahan konvensional yang memerlukan tiga kali proses untuk menerjemahkan ketiga data tersebut.

Jika MULTI-WD ini digunakan oleh pengguna atau komunitas Wikidata secara masif, kelengkapan multibahasa pada data Wikidata akan meningkat. Hal ini dapat memberikan kontribusi berupa kelengkapan data pada Wikidata secara multibahasa untuk penelitian di bidang sistem KGQA berbasis Wikidata.

V. KESIMPULAN

MULTI-WD dapat digunakan untuk melakukan pemprofilan data Wikidata secara multibahasa. MULTI-WD ini dapat menjadi jawaban atas permasalahan isu ketidaklengkapan data Wikidata dari sisi multibahasa yang saat ini belum tersedia penelitiannya. Penerapan MULTI-WD dapat memberikan kontribusi berupa kelengkapan *dataset* yang digunakan pada penelitian di bidang sistem KGQA berbasis Wikidata. MULTI-WD telah diuji dengan menggunakan pendekatan *black box* untuk empat fungsi utama, yaitu pemilihan kategori, penampilan statistik multibahasa, penerjemahan data, dan pembaruan data ke Wikidata oleh lima responden dari komunitas Wikidata. Hasil pengujian menunjukkan bahwa semua fungsi MULTI-WD dapat berjalan 100%. Sementara itu, dari sisi efisiensi, MULTI-WD dapat melakukan proses penerjemahan data Wikidata sebesar 300%. Dari sisi kesempurnaan hasil terjemahan, MULTI-WD masih memiliki keterbatasan dalam hal tata bahasa yang sesuai dengan bahasa manusia. Pada penelitian mendatang, MULTI-WD dapat

dikembangkan dengan menggunakan model bahasa terbuka yang lebih baik, sesuai dengan *state-of-the-art* yang ada, dengan tetap memperhatikan sumber daya yang tidak terlalu mahal. Selain itu, MULTI-WD dapat digunakan untuk melakukan pemprofilan bahasa lain sesuai dengan perkembangan *dataset* yang ada untuk sistem KGQA.

KONFLIK KEPENTINGAN

Dengan ini para penulis menyatakan bahwa tidak ada konflik kepentingan.

KONTRIBUSI PENULIS

Konseptualisasi, Mohammad Yani; metodologi, Mohammad Yani; penulisan—penyusunan draf asli, Mohammad Yani dan Lilyan Arhatia Agustine; penulisan—peninjauan dan penyuntingan, Mohammad Yani; perangkat lunak, Lilyan Arhatia Agustine; visualisasi, Lilyan Arhatia Agustine; pengawasan, Mohammad Yani dan Iryanto; administrasi proyek, Lilyan Arhatia Agustine; akuisisi pendanaan, Mohammad Yani; sumber daya, Mohammad Yani dan Lilyan Arhatia Agustine; validasi, Mohammad Yani dan Lilyan Arhatia Agustine; investigasi, Mohammad Yani dan Iryanto.

UCAPAN TERIMA KASIH

Ucapan terima kasih disampaikan kepada Pusat Penelitian dan Pengabdian Kepada Masyarakat Politeknik Negeri Indramayu atas dana penelitian yang telah diberikan. Penelitian ini merupakan luaran tambahan dari kegiatan penelitian yang dibiayai oleh Pusat Penelitian dan Pengabdian Kepada Masyarakat Politeknik Negeri Indramayu dengan nomor kontrak: 0720/PL42.PL42.9/AL.04/2024.

REFERENSI

- [1] A. Hogan dkk., “Knowledge graphs,” *ACM Comput. Surv.*, vol. 54, no. 4, hal. 1-37, Mei 2022, doi: 10.1145/3447772.
- [2] A.A. Krisnadhi, M. Yani, dan I. Budi, “Entity and relation linking for knowledge graph question answering using gradual searching,” *J. Nas. Tek. Elek. Teknol. Inf.*, vol. 13, no. 2, hal. 139-146, Mei 2024, doi: 10.22146/jnteti.v13i2.9184.
- [3] M. Yani dan A.A. Krisnadhi, “Challenges, techniques, and trends of simple knowledge graph question answering: A survey,” *Information*, vol. 12, no. 7, hal. 1-31, Jul. 2021, doi: 10.3390/info12070271.
- [4] S. Figueroa, “Knowledge discovery in Wikidata with machine learning in graph,” dalam *Inf. Syst. Technol.*, A. Rocha dkk., Eds., Cham, Swiss: Springer, 2024, hal. 3-12, doi: 10.1007/978-3-031-45645-9_1.
- [5] K. Tharani, “Much more than a mere technology: A systematic review of Wikidata in libraries,” *J. Acad. Librariansh.*, vol. 47, no. 2, hal. 1-8, Mar. 2021, doi: 10.1016/j.acalib.2021.102326.
- [6] K. Shenoy dkk., “A study of the quality of Wikidata,” *J. Web Semant.*, vol. 72, hal. 1-10, Apr. 2022, doi: 10.1016/j.websem.2021.100679.
- [7] D. Vrandečić dan M. Krötzsch, “Wikidata: A free collaborative knowledgebase,” *Commun. ACM*, vol. 57, no. 10, hal. 78-85, Okt. 2014, doi: 10.1145/2629489.
- [8] M.U. Akhtar dkk., “Entity alignment based on relational semantics augmentation for multilingual knowledge graphs,” *Knowl.-Based Syst.*, vol. 252, hal. 1-10, Sep. 2022, doi: 10.1016/j.knosys.2022.109494.
- [9] A. Perevalov, D. Diefenbach, R. Usbeck, dan A. Both, “QALD-9-plus: A multilingual dataset for question answering over DBpedia and Wikidata translated by native speakers,” dalam *2022 IEEE 16th Int. Conf. Semant. Comput. (ICSC)*, 2022, hal. 229-234, doi: 10.1109/ICSC52841.2022.00045.
- [10] Z. Shaik, F. Ilievski, dan F. Morstatter, “Analyzing race and country of citizenship bias in Wikidata,” dalam *2021 IEEE 18th Int. Conf. Mob. Ad Hoc Smart Syst. (MASS)*, 2021, hal. 665-666, doi: 10.1109/MASS52906.2021.00099.
- [11] A. Pratapa, R. Gupta, dan T. Mitamura, “Multilingual event linking to Wikidata,” dalam *Proc. Workshop Multiling. Inf. Access (MIA)*, 2022, hal. 37-58, doi: 10.18653/v1/2022.mia-1.5.

- [12] F. Darari, R.E. Prasojo, S. Razniewski, dan W. Nutt, "COOL-WD: A completeness tool for Wikidata," dalam *CEUR Workshop Proc.*, 2017, hal. 1–4.
- [13] A. Wisesa dkk., "Wikidata completeness profiling using ProWD," dalam *K-CAP 2019 - Proc. 10th Int. Conf. Knowl. Capture*, 2019, hal. 123–130, doi: 10.1145/3360901.3364425.
- [14] L.-A. Kaffee dkk., "Multilingual knowledge graphs and low-resource languages: A review", *Trans. Graph Data Knowl. (TGDK)*, vol. 1, No. 1, hal. 1–19, Des. 2023, doi: 10.4230/TGDK.1.1.10.
- [15] M. Yani, A.A. Krisnadhi, dan I. Budi, "A better entity detection of question for knowledge graph question answering through extracting position-based patterns," *J. Big Data*, vol. 9, hal. 1–26, Jun. 2022, doi: 10.1186/s40537-022-00631-1.
- [16] E. Prud'hommeaux dan A. Seaborne. "SPARQL query language for RDF." W3C. Tanggal akses: 15-Jan-2024. [Online]. Tersedia: <https://www.w3.org/TR/rdf-sparql-query/>
- [17] G. Xiao dan J. Corman, "Ontology-mediated SPARQL query answering over knowledge graphs," *Big Data Res.*, vol. 23, hal. 1–25, Feb. 2021, doi: 10.1016/j.bdr.2020.100177.
- [18] M. Mosser dkk., "Querying APIs with SPARQL," *Inf. Syst.*, vol. 105, hal. 1–14, Mar. 2022, doi: 10.1016/j.is.2020.101650.
- [19] M. Bakhshi, M. Nematbakhsh, M. Mohsenzadeh, dan A.M. Rahmani, "Data-driven construction of SPARQL queries by approximate question graph alignment in question answering over knowledge graphs," *Expert Syst. Appl.*, vol. 146, hal. 1–19, Mei 2020, doi: 10.1016/j.eswa.2020.113205.
- [20] K. Syamsi dkk., "Developing a culture-based Indonesian language textbook for non-native speakers for academic purposes," *Cakrawala Pendidik.*, vol. 43, no. 1, hal. 115–126, Feb. 2024, doi: 10.21831/cp.v43i1.60321.
- [21] O. Bojar dkk., "Findings of the 2018 conference on machine translation (WMT18)," dalam *Proc. 3rd Conf. Mach. Transl. (WMT)*, 2018, hal. 272–303, doi: 10.18653/v1/W18-64028.