Terjemahan dari 10.22146/jnteti.v13i3.11580

Named Entity Recognition pada Kueri Pencarian Statistik

Wildannissa Pinasti[1], Lya Hulliyyatus Suadaa[1]

[1] Program Studi Komputasi Statistik, Politeknik Statistika STIS, Jakarta Timur, DKI Jakarta 13330, Indonesia

[Diserahkan: 20 Januari 2024, Direvisi: 12 April 2024, Diterima: 10 Juli 2024]
Penulis Korespondensi: Lya Hulliyyatus Suadaa (email: lya@stis.ac.id)

INTISARI — Mesin pencarian perlu memahami keinginan pengguna untuk mengembalikan hasil pencarian yang relevan. Salah satu teknik yang dapat digunakan untuk lebih memahami tujuan pengguna adalah identifikasi entitas dalam kueri menggunakan *named entity recognition* (NER). Mengetahui jenis entitas pada kueri dapat menjadi langkah awal dalam membantu mesin pencari memahami lebih baik tujuan pencarian. Dalam penelitian ini, sebuah *dataset* dibangun menggunakan riwayat kueri pencarian dari situs web Badan Pusat Statistik (BPS) dan pemodelan NER pada kueri dilakukan untuk mengekstrak entitas pada kueri pencarian yang terkait dengan data statistik. Tahapan dalam penelitian ini meliputi pengumpulan data kueri, prapemrosesan data kueri, pelabelan data kueri, pengembangan model NER pada kueri, dan evaluasi model. Model *conditional random field* (CRF) digunakan untuk pemodelan NER pada kueri dengan dua skenario: CRF dengan fitur dasar dan CRF dengan fitur dasar ditambah dengan fitur *part of speech* (POS). Model CRF digunakan karena efektivitasnya yang terkenal dalam *natural language processing* (NLP), terutama untuk tugas seperti NER dengan pelabelan urutan kata. Pada penelitian ini, model dasar CRF dan CRF dengan penambahan fitur POS mencapai *F*1-*score* sebesar 0,9139 dan 0,9110 berturut-turut. Sebuah studi kasus tentang pencarian pada Linked Open Data (LOD) untuk *dataset* statistik menunjukkan bahwa pencarian dengan kueri yang telah melewati NER dan menggunakan ekspansi kueri sinonim memberikan hasil pencarian yang lebih baik dibandingkan dengan pencarian tanpa NER dan pencarian dengan NER tanpa ekspansi kueri. Kinerja model yang menggunakan fitur tambahan *POS tagging* tidak menghasilkan peningkatan yang signifikan. Oleh karena itu, disarankan agar penelitian di masa depan memperdalam penggunaan *deep learning*.

KATA KUNCI — *Named Entity Recognition*, Kueri, Pencarian *Dataset*, *Conditional Random Fields*, Linked Open Data.

I. PENDAHULUAN

AllStat adalah aplikasi pencarian yang digunakan di situs web Badan Pusat Statistik (BPS). Aplikasi pencarian ini memproses jutaan kueri untuk membantu dalam penyediaan layanan data statistik di BPS, seperti mengakses *dataset*, infografis, dan publikasi statistik. Salah satu tantangan mendasar pada mesin pencari adalah memahami kueri yang ambigu dalam pencarian untuk memberikan hasil pencarian yang relevan. Referensi [1] mendefinisikan kueri ambigu sebagai kueri yang memiliki lebih dari satu makna. Kueri ambigu juga dapat terjadi karena entitas dalam kueri memiliki banyak referensi lain, seperti alias, singkatan, dan ejaan alternatif [2]. Sebagai contoh, entitas "jumlah penduduk" dapat mengambil bentuk lain seperti "populasi" atau bahkan "jumlah warga" sebagai istilah alternatif.

Berdasarkan wawancara dengan pakar di BPS, fitur-fitur mesin pencarian BPS diharapkan dapat melakukan pencarian pada Linked Open Data (LOD) untuk *dataset* statistik dan memahami kueri pengguna dengan lebih baik, bahkan jika istilah yang digunakan ambigu. Istilah *linked data* merujuk pada seperangkat prinsip untuk mempublikasikan dan menghubungkan data terstruktur di web dengan cara yang dapat dibaca oleh mesin [3]. *Linked data* yang dipublikasikan di bawah lisensi terbuka dan secara kolektif disebut sebagai LOD.

Tugas tersebut dapat diselesaikan menggunakan *named entity recognition* (NER) [4]. NER bertujuan untuk mengidentifikasi dan mengklasifikasikan entitas dari teks ke dalam kategori-kategori yang telah ditentukan sebelumnya, seperti orang, lokasi, dan organisasi [5]. NER tidak hanya berfungsi sebagai alat untuk *information extraction* (IE), tetapi juga memainkan peran penting dalam berbagai aplikasi *natural language processing* (NLP), seperti memahami teks [6], menjawab pertanyaan [7], dan mengambil informasi [8].

NER merupakan salah satu tugas NLP yang biasanya dilakukan pada dokumen teks. Namun, model NER yang dilatih pada kalimat-kalimat panjang dan gramatikal sering kali kesulitan untuk bekerja dengan baik pada kueri karena kueri memiliki karakteristik yang berbeda dibandingkan dengan teks pada umumnya [9]. Kueri biasanya terdiri atas jumlah kata yang sedikit dan kurang memiliki konteks. Sebuah studi tentang analisis log kueri pencarian *dataset* pada portal data menunjukkan bahwa hampir 90% kueri terdiri atas satu hingga tiga kata, dengan rata-rata panjang kueri sebesar 2,67 kata [10].

Untuk mengatasi tantangan ini, sebuah studi membahas masalah baru dalam pencarian web, yang disebut *named entity recognition in query* (NERQ), yang bertujuan untuk mendeteksi entitas dalam kueri dan mengategorikannya ke dalam kelas-kelas [8]. Studi tersebut mengusulkan pendekatan pemodelan topik *weakly supervised latent dirichlet allocation* (WS-LDA) menggunakan *dataset* yang secara khusus diberi label. Hasil analisis dari penelitian ini juga menunjukkan bahwa sekitar 71% dari kueri pencarian mengandung entitas bernama, yang menunjukkan bahwa identifikasi entitas bernama dalam kueri dapat membantu memahami tujuan pencarian pengguna dengan lebih baik.

Sebuah studi lain melakukan NER pada kueri pencarian yang berkaitan dengan travel menggunakan pembelajaran mesin untuk mengekstrak entitas dari kueri [11]. Studi ini menerapkan model *conditional random fields* (CRF) dengan *dataset* yang diberi label secara manual dan mencapai hasil akurasi yang tinggi. Penelitian ini menunjukkan bahwa metode

NLP secara efektif dapat diterapkan untuk NER pada kueri dengan domain spesifik [11].

Metode berbasis aturan merupakan aturan yang banyak digunakan dalam pengembangan awal NER. Metode sederhana dan efektif ini bergantung pada pembentukan basis pengetahuan dan kamus yang mengakibatkan biaya pemeliharaan yang tinggi [12]. Meskipun studi awal NER sebagian besar mengandalkan metode berbasis aturan, studi terkini telah beralih ke pembelajaran terbimbing. Beberapa teknik pembelajaran terbimbing yang dapat digunakan untuk NER termasuk *hidden Markov models* (HMM) dan CRF, dengan algoritma CRF menunjukkan kinerja yang lebih baik daripada HMM [5]. CRF adalah model probabilistik yang banyak diakui berguna untuk berbagai tugas NLP, terutama tugas pelabelan urutan seperti NER [13].

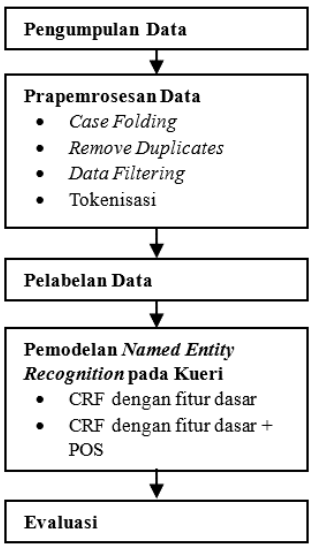
Di Indonesia, NER untuk bahasa Indonesia telah dikembangkan secara luas. Satu studi melakukan NER pada artikel berita dengan 15 kelas entitas, melampaui jumlah kelas dalam NER Indonesia yang sudah ada [14]. NER juga telah diterapkan dalam berbagai aplikasi menggunakan bahasa Indonesia, seperti mengekstraksi informasi kondisi lalu lintas di kota pada waktu tertentu [15], mengekstraksi informasi terkait gangguan listrik [16], dan mendapatkan informasi yang relevan untuk wisatawan dari ulasan destinasi wisata [17]. Studi lain berfokus pada NER dalam teks yang lebih pendek, seperti NER dalam cuitan berbahasa Indonesia menggunakan CRF [18]. Namun, belum ada penelitian tentang NER dalam kueri berbahasa Indonesia. Telah diakui bahwa metode yang sensitif terhadap bahasa dalam NLP lebih akurat daripada metode yang tidak bergantung pada bahasa [19]. Secara khusus, bahasa Indonesia memiliki morfologi derivasional yang kaya, termasuk pengulangan [20]. Menangkap morfologi melalui *dataset* yang spesifik terhadap bahasa dapat meningkatkan kinerja NER, termasuk NER dalam kueri bahasa Indonesia.

Berdasarkan latar belakang tersebut, penelitian ini bertujuan untuk memodelkan NER pada kueri untuk mengekstrak entitas dalam kueri pencarian statistik. Kontribusi yang dilakukan disajikan sebagai berikut.

1. Membangun *dataset* untuk NER pada kueri pencarian yang terkait dengan *dataset* statistik menggunakan data riwayat pencarian dari situs web BPS.
2. Mengembangkan model NER pada kueri dengan menggunakan model CRF.
3. Melakukan sebuah studi kasus tentang pencarian pada LOD yang berisi *dataset* statistik BPS menggunakan kueri-kueri yang telah diiedentifikasi jenis entitas mereka melalui NER pada kueri.

II. METODOLOGI

Penelitian ini berfokus pada analisis kueri yang terkait dengan pencarian *dataset* statistik dalam bidang statistik sosiodemografi untuk periode tahun 2020 hingga 2022. Tahapan dalam penelitian ini meliputi pengumpulan data kueri, prapemrosesan data kueri, pelabelan data kueri untuk membentuk sebuah *dataset*, pengembangan model NER dalam kueri, dan evaluasi model. Alur penelitian diperlihatkan pada Gambar 1.



Gambar 1. Alur penelitian.

A. PENGUMPULAN DATA

Data yang digunakan dalam penelitian ini adalah kueri pencarian pengguna dari aplikasi AllStat yang digunakan di situs web oleh BPS. Data yang dikumpulkan mencakup data kueri dari tahun 2020 hingga 2022 dengan atribut seperti kueri, tahun, total pencarian bulanan, dan total pencarian dalam satu tahun.

B. PRAPEMROSESAN DATA

Prapemrosesan data adalah proses persiapan data yang dikumpulkan untuk pemrosesan atau analisis lebih lanjut pada tahap-tahap berikutnya [21]. Dalam prapemrosesan terdapat beberapa tahapan sebagai berikut.

1) CASE FOLDING

Case folding dilakukan untuk mengonversi semua huruf pada data kueri menjadi huruf kecil atau *lowercase*. Pada data kueri yang diperoleh, kueri dicatat dengan aturan *case sensitive*, sehingga perlu dilakukan *case folding* menjadi huruf kecil untuk menyamakan representasi teks agar tidak ada perbedaan antara huruf besar dan huruf kecil dalam pemrosesan teks.

2) REMOVE DUPLICATES

Remove duplicates merupakan proses yang dilakukan untuk mengidentifikasi dan menghapus kueri yang sama. Pada data kueri yang diperoleh, kueri dan jumlah kemunculannya dicatat berdasarkan tahun, sehingga memungkinkan adanya kueri yang sama pada tahun berbeda. Selain itu, hasil *case folding* pada tahap sebelumnya juga memungkinkan adanya kueri yang awalnya dicatat berbeda menjadi sama karena sudah menjadi huruf kecil. Oleh karena itu, proses *remove duplicates* dilakukan untuk mendapatkan kueri yang unik.

3) DATA FILTERING

Tahap *data filtering* dilakukan untuk menghilangkan kueri yang tidak relevan pada studi ini, yaitu kueri yang tidak terkait dengan pencarian *dataset* statistik sosiodemografi. Dalam *data filtering*, data kueri yang telah melalui tahap prapemrosesan sebelumnya akan difilter secara semiotomatis. Pemfilteran otomatis dilakukan dengan membuat aturan, misalnya dengan membuat daftar kata-kata yang tidak relevan yang biasanya muncul dalam kueri yang tidak relevan untuk menghapus kueri yang mengandung kata-kata tersebut.

4) TOKENISASI

Sebelum dilabeli, data kueri yang telah tersaring dikenai proses tokenisasi terlebih dahulu. Tokenisasi adalah proses membagi kalimat menjadi token atau bagian-bagian tertentu.

C. PELABELAN DATA

Seperti yang dilakukan dalam beberapa penelitian sebelumnya [11], [22], kelas entitas untuk pelabelan dapat

disesuaikan dengan domain. Kelas label dalam penelitian ini ditentukan berdasarkan domain statistik, menyesuaikan konsep yang digunakan di BPS. Label kelas entitas yang digunakan dalam penelitian ini dijelaskan sebagai berikut.

1) *STATISTICAL INDICATOR* (SI)

Statistical indicator (SI) mendeskripsikan karakteristik ekonomi, sosial, dan fenomena lainnya pada waktu dan tempat tertentu. Misalnya, "jumlah penduduk".

2) *STATISTICAL CLASSIFICATION* (SC)

Sebuah set kategori yang mewakili nilai satu atau lebih variabel yang tercantum dalam survei statistik atau data administratif digunakan dalam proses dan penyebaran statistik. Misalnya, "jenis kelamin", "provinsi".

3) *CLASSIFICATION ITEM* (CI)

Classification item (CI) adalah sebuah set kategori pada tingkat tertentu dalam klasifikasi statistik yang mendefinisikan isi dan batas-batas kategori tersebut. Contohnya, "perempuan", "DKI Jakarta".

4) PERIODE (P)

Periode (P) mewakili periode waktu dari data statistik yang hendak dicari oleh pengguna. Contohnya, 2020.

Proses pelabelan dilakukan menggunakan notasi *begin*, *inside*, *outside* (BIO). Untuk setiap entitas yang teridentifikasi ke dalam kelas entitas X, token di awal entitas tersebut diberi label sebagai B-X. Jika entitas yang dikenali terdiri atas dua atau lebih token, token kedua dan seterusnya dalam entitas tersebut diberi label sebagai I-X. Token yang tidak termasuk dalam kelas entitas apa pun diberi label sebagai O.

Proses pelabelan dilakukan secara semiotomatis. Pelabelan otomatis menggunakan beberapa metode untuk mempercepat proses pelabelan dan mengurangi beban pada *annotator* manusia. Selanjutnya, pelabelan manual dilakukan pada token yang belum dilabeli dalam pelabelan otomatis dan untuk memperbaiki hasil dari pelabelan otomatis. Pelabelan otomatis dilakukan menggunakan beberapa aturan sebagai berikut.

1. Tahun diklasifikasikan sebagai kelas entitas periode dengan label P. Pelabelan tahun sebagai periode dilakukan pada token-token yang terdiri atas empat digit menggunakan bantuan RegEx.
2. Nama-nama daerah diklasifikasikan sebagai kelas entitas CI dengan label CI. Dalam pelabelan ini, digunakan daftar nama provinsi dan kabupaten/kota di Indonesia, beserta sinonim dan singkatan-singkatannya. Token dalam kueri akan diberi label sebagai CI jika cocok dengan daftar kata tersebut.
3. Jenis kelamin diklasifikasikan sebagai kelas entitas CI dengan label CI. Dalam pelabelan ini, digunakan daftar jenis kelamin beserta sinonim-sinonimnya.
4. Nama-nama indikator statistik diklasifikasikan sebagai kelas entitas SI dengan label SI. Dalam pelabelan ini, digunakan daftar indikator statistik di bidang sosiodemografi dari BPS.
5. *Stopword*, seperti kata penghubung "menurut", tidak diklasifikasikan ke dalam kelas entitas apa pun, sehingga diberi label sebagai *outside* dengan label O. Pelabelan ini menggunakan daftar *stopword* dalam bahasa Indonesia dari perpustakaan Python Sastrawi.

Pelabelan manual dilakukan pada token-token yang tersisa yang belum dilabeli oleh dua *annotator* yang dilengkapi dengan pedoman pelabelan. Untuk mengukur kualitas pelabelan kueri, keandalan *inter-rater* dihitung menggunakan metode *Krippendorff's alpha* [23].

Selain pelabelan kelas entitas untuk NER, *part of speech tagging (POS tagging)* juga dilakukan untuk meningkatkan kinerja model seperti yang ditunjukkan dalam beberapa upaya penelitian sebelumnya dalam NER umum menggunakan CRF [24]. *POS tagging* adalah proses untuk mengategorikan kata-kata ke dalam kelas-kelas berdasarkan posisi kata-kata tersebut dalam sebuah kalimat [25]. Kelas kata ini dapat mencakup kata kerja, kata sifat, kata benda, dan sebagainya. Hasil dari *POS tagging* digunakan sebagai fitur tambahan dengan tujuan untuk meningkatkan kinerja model NER CRF. Pustaka Flair dalam Python digunakan untuk *POS tagging*.

D. PEMODELAN NER PADA KUERI

Pemodelan NER dalam kueri di penelitian ini dilakukan menggunakan model *linear chain CRF* (LC-CRF) untuk mengidentifikasi jenis entitas dalam kueri [26]. CRF adalah model probabilistik yang diakui berguna untuk berbagai tugas NLP, terutama tugas pelabelan urutan seperti NER [13].

CRF termasuk dalam kategori model grafis tak berarah yang dapat digunakan untuk menghitung *conditional probability* dari keluaran runtun $y = <y_1, \ldots, y_T>$ dengan diberikan sebuah masukan runtun $x = <x_1, \ldots, x_T>$. Z adalah faktor normalisasi untuk runtun keadaan. Dalam NER pada kueri, masukan x adalah urutan token dalam sebuah kueri, sedangkan keluaran y adalah label urutan yang mewakili segmentasi kode dan informasi kelas entitas, misalnya B-SI untuk merepresentasikan awalan entitas untuk SI. Sementara itu, λk adalah nilai bobot yang dipelajari untuk fk.

$$P(y|x) = \frac{1}{Z} exp \sum_{t=1}^{T} \sum_{k=1}^{K} \lambda_k f_k(y_t y_{t-1}, x) \quad (1)$$

E. EVALUASI

NER dievaluasi pada data uji dengan membandingkan keluaran dengan pelabelan manusia sebagaimana diukur oleh perhitungan presisi, *recall*, dan *F*1-*score* [27]. Presisi, *recall*, dan *F*1-*score* dihitung berdasarkan jumlah *true positive* (TP), *false positive* (FP), dan *false negative* (FN)

1. TP terjadi ketika entitas yang dikenali oleh NER sesuai dengan *ground truth*.
2. FP terjadi ketika entitas yang dikenali oleh NER tidak sesuai dengan *ground truth*.
3. FN terjadi ketika entitas yang dilabeli dalam *ground truth* tidak dikenali oleh NER.

Presisi, *recall*, dan *F*1-*score* dihitung sebagai berikut.

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

$$F1\ score = \frac{Precision\ .Recall}{Precision + Recall}. \quad (4)$$

F. STUDI KASUS: PENCARIAN PADA LINKED OPEN DATA (LOD) TABEL DINAMIS BPS

Setelah membangun *dataset* dan melakukan NER pada kueri, sebuah studi kasus dari kueri yang telah melewati tahap NER dilakukan untuk mencari pada LOD untuk *dataset* statistik BPS. Tujuan studi kasus ini adalah untuk memberikan gambaran tentang penggunaan kueri yang telah melalui proses NER dalam membantu menyediakan konteks. Dalam penelitian ini, pencarian dilakukan pada LOD, khususnya untuk kueri-kueri dengan istilah-istilah yang ambigu.

TABEL I
HASIL PELABELAN PADA TINGKAT TOKEN

Label Token	Jumlah Token
B-SI	3.219
I-SI	5.020
B-SC	505
I-SC	321
B-CI	1.921
I-CI	1.022
B-P	3.070
I-P	64
O	3.054
Jumlah keseluruhan	18.196

TABEL II
HASIL PELABELAN PADA TINGKAT ENTITAS

Label Entitas	Jumlah Entitas
SI	3.219
SC	505
CI	1.921
P	3.070
O	3.054
Jumlah keseluruhan	11.769

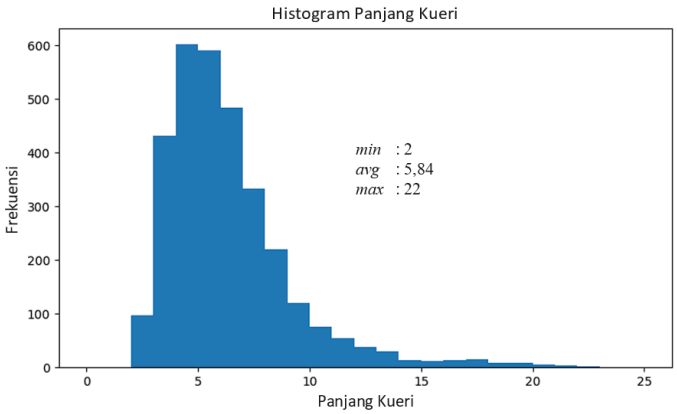
Namun, tidak semua entitas dalam kueri yang diidentifikasi oleh proses NER menggunakan istilah standar yang digunakan BPS. Sebagai contoh, BPS menggunakan istilah "jumlah penduduk" sebagai SI untuk data yang menggambarkan jumlah penduduk di suatu wilayah. Namun, terdapat kueri-kueri yang menggunakan istilah "populasi" untuk menanyakan data yang sama. Di sisi lain, LOD dibangun dengan entitas-entitas standar yang mengacu pada *metadata management system* (MMS) di BPS. Perbedaan istilah antara entitas yang disebutkan oleh pengguna dan yang ada di LOD menyebabkan diperlukannya proses untuk menghubungkan entitas-entitas ini guna mencapai hasil pencarian yang sesuai dengan tujuan pengguna.

Untuk mengatasi masalah tersebut, penelitian ini melakukan ekspansi dengan mencari sinonim untuk entitas-entitas yang ambigu, sehingga pencarian tidak hanya mencakup istilah yang disebutkan, tetapi juga sinonimnya. Pencarian kata sinonim dilakukan menggunakan Tesaurus Bahasa Indonesia. Dalam penelitian ini, tiga sinonim dengan kesamaan tertinggi dipilih untuk perluasan kueri. Selanjutnya, pencarian data LOD dilakukan menggunakan SPARQL. SPARQL merupakan bahasa kueri dan protokol standar untuk LOD di web atau untuk data dengan model RDF [28]. Studi kasus pencarian pada LOD dilakukan dengan membandingkan pencarian kueri dengan perluasan sinonim dan kueri asli tanpa perluasan sinonim. Selanjutnya, evaluasi manusia dilakukan untuk membandingkan kedua hasil pencarian tersebut.

III. HASIL DAN PEMBAHASAN

A. DATASET

Data yang diperoleh untuk penelitian ini terdiri atas 2.324.645 kueri. Namun, kueri-kueri ini perlu melalui tahap prapemrosesan sebelum siap untuk dilabeli. Prapemrosesan dimulai dengan mengonversi semua kata dalam kueri menjadi huruf kecil untuk standardisasi representasi kueri. Setelah itu, duplikat dihapus dan penyaringan semiotomatis dilakukan untuk mendapatkan kueri-kueri yang relevan, menghasilkan 3.145 kueri. Terhadap kueri-kueri ini kemudian dilakukan tokenisasi, menghasilkan 18.196 token yang siap untuk dilabeli.



Gambar 2 Histogram panjang kueri.

TABEL III
CONTOH HASIL PELABELAN

Kueri	Token	Label
jumlah penduduk dki jakarta menurut kelompok umur tahun 2020	jumlah	B-SI
	penduduk	I-SI
	dki	B-CI
	jakarta	I-CI
	menurut	O
	kelompok	B-SC
	umur	I-SC
	tahun	O
	2020	B-P

Pelabelan dilakukan secara semiotomatis pada 18.196 token. Jumlah total token yang dilabeli secara otomatis adalah 12.372 token, yang mencakup 68,04% dari keseluruhan token. Kemudian, pelabelan manual dilakukan oleh dua *annotator*, termasuk pemeriksaan hasil pelabelan otomatis. Dalam penelitian ini, nilai *alpha* sebesar 0,6515 diperoleh, menunjukkan tingkat kesepakatan *annotator* yang cukup. Distribusi hasil pelabelan pada tingkat token dapat dilihat di Tabel I dan distribusi pada tingkat entitas dapat dilihat di Tabel II.

Gambar 2 memperlihatkan distribusi panjang kueri dalam *dataset*. Panjang rata-rata kueri adalah lima hingga enam kata, dengan kueri terpendek terdiri atas dua kata dan kueri terpanjang ditemukan dalam kueri dengan 22 kata.

Contoh dari *dataset* yang telah dilabeli dapat dilihat pada Tabel III. Pada contoh tersebut, frasa "jumlah penduduk" dalam kueri diidentifikasi sebagai entitas yang termasuk ke dalam kelas SI. Oleh karena itu, kata "jumlah" dilabeli sebagai B-SI (*begin*), yang menunjukkan awal dari sebuah entitas SI, sedangkan kata "penduduk" yang mengikutinya dilabeli sebagai I-SI (*inside*), yang menunjukkan bahwa kata tersebut berada di dalam entitas SI. Pola pelabelan yang sama berlaku untuk entitas yang diakui lainnya, dengan setiap entitas yang diakui dimulai dengan label *begin* dan setiap kata berikutnya dalam entitas yang sama dilabeli sebagai *inside*. *Outside* digunakan untuk melabeli kata-kata yang tidak termasuk ke dalam kelas entitas apa pun.

Selain pelabelan entitas menggunakan NER, penelitian ini juga melakukan *POS tagging* sebagai fitur tambahan dalam model. Untuk *POS tagging*, digunakan pustaka Flair dalam bahasa pemrograman Python. Model dilatih menggunakan korpus yang disediakan oleh Flair. Pelatihan dilakukan di Google Colaboratory dan model mencapai *F*1-*score* sebesar 93,59% pada data uji. Gambar 3 menunjukkan kinerja model dalam mengenali setiap kelas. Model dapat memprediksi SYM

	precision	recall	f1-score	support
NOUN	0.9008	0.9112	0.9060	2511
PROPN	0.9276	0.9246	0.9261	2162
PUNCT	0.9969	1.0000	0.9985	1623
VERB	0.9562	0.9372	0.9466	1258
ADP	0.9524	0.9515	0.9520	1114
PRON	0.9532	0.9798	0.9663	644
ADJ	0.8279	0.8279	0.8279	488
NUM	0.9765	0.9740	0.9752	384
CCONJ	0.9836	0.9945	0.9890	362
DET	0.9574	0.9238	0.9403	341
ADV	0.8896	0.8150	0.8507	346
AUX	0.9463	1.0000	0.9724	229
SCONJ	0.8300	0.8557	0.8426	194
PART	0.9149	0.9663	0.9399	89
SYM	1.0000	1.0000	1.0000	6
X	0.0000	0.0000	0.0000	5
accuracy			0.9359	11756
macro avg	0.8758	0.8788	0.8771	11756
weighted avg	0.9355	0.9359	0.9356	11756

Gambar 3. Hasil Pemodelan *POS tagging*.

query_id	query	token_no	token	postag	label
767	jumlah penduduk indonesia 2020	1	jumlah	NOUN	B-SI
767	jumlah penduduk indonesia 2020	2	penduduk	NOUN	I-SI
767	jumlah penduduk indonesia 2020	3	indonesia	PROPN	B-CI
767	jumlah penduduk indonesia 2020	4	2020	NUM	B-P
1311	ipm 2020	1	ipm	PROPN	B-SI
1311	ipm 2020	2	2020	NUM	B-P
48185	angka morbiditas tahun 2021	1	angka	NOUN	B-SI
48185	angka morbiditas tahun 2021	2	morbiditas	NOUN	I-SI
48185	angka morbiditas tahun 2021	3	tahun	NOUN	O
48185	angka morbiditas tahun 2021	4	2021	NUM	B-P
607401	bandung 2010 populasi	1	bandung	NOUN	B-CI
607401	bandung 2010 populasi	2	2010	NUM	B-P
607401	bandung 2010 populasi	3	populasi	NOUN	B-SI

Gambar 4. Gambaran dari *dataset*.

dan PUNCT dengan kinerja terbaik di antara semua *POS tagging*, dengan *F*1-*score*, secara berturut-turut, 100,00% dan 99,85%. Model kemudian digunakan untuk memprediksi *tag* untuk semua token. Namun, model gagal memprediksi *tag* X karena kinerja yang dihasilkan untuk *tag* ini adalah 0%. Kinerja prediksi rendah juga ditunjukkan untuk *tag* ADJ, yang mencapai kinerja sebesar 82,79%. Model kemudian digunakan untuk memprediksi *POS tagging* untuk semua token. Gambaran dari *dataset* yang telah dibangun disediakan dalam Gambar 4. Gambar tersebut menunjukkan beberapa baris dari *dataset*, menyajikan informasi tentang "query", "token", "postag", dan "label".

B. PEMODELAN NER PADA KUERI DAN EVALUASI

Pemodelan NER dengan CRF dilakukan menggunakan pustaka CRFsuite dalam bahasa pemrograman Python. Dua skenario digunakan, yaitu CRF dengan fitur dasar dan CRF dengan fitur dasar dan POS, yang selanjutnya disebut sebagai CRF POS. Fitur dasar yang digunakan dalam pemodelan adalah pada tingkat kata, termasuk kata itu sendiri, fitur sufiks, dan bentuk kata, merupakan digit atau tidak. Studi ini tidak menggunakan fitur *case* karena data sudah dalam huruf kecil.

Setiap skenario model CRF menggunakan *hyperparameter* berdasarkan hasil *grid search*. Tabel IV menunjukkan hasil eksperimen yang telah dilakukan menggunakan 5-*fold cross-validation*. Berdasarkan hasil kinerja, model CRF dengan fitur dasar mengungguli model CRF dengan fitur POS tambahan, meskipun perbedaan dalam *F*1-*score* tidak signifikan. *F*1-*score* untuk CRF dengan fitur dasar dan CRF POS adalah 0,9139 dan 0,9110, secara berturut-turut. Kinerja yang lebih rendah dari model CRF dengan fitur POS dapat dikaitkan dengan sifat

TABEL IV
KINERJA MODEL NER PADA KUERI

Model	Presisi	*Recall*	*F1-Score*
CRF dengan fitur dasar	0,9249	0,9031	0,9139
CRF POS	0,9215	0,8999	0,9110

TABEL V
KINERJA MODEL CRF DENGAN FITUR DASAR

Entitas	Presisi	*Recall*	*F1-Score*
SI	0,8860	0,8668	0,8763
SC	0,8560	0,7553	0,8020
CI	0,9075	0,8745	0,8903
P	0,9863	0,9844	0,9854

TABEL VI
KINERJA MODEL CRF DENGAN FITUR DASAR + POS

Entitas	Presisi	*Recall*	*F1-Score*
SI	0,8795	0,8625	0,8709
SC	0,8402	0,7515	0,7929
CI	0,9102	0,8709	0,8896
P	0,9869	0,9831	0,9850

singkat dan tidak lengkap dari kueri-kueri. Selain itu, 60% dari label *POS tagging* dalam *dataset* adalah NOUN, yang tidak signifikan dalam meningkatkan pembelajaran model NER pada kueri.

Selain kinerja model secara keseluruhan, kinerja model NER juga dapat dilihat pada tingkat entitas. Tabel V menunjukkan kinerja model fitur dasar CRF untuk setiap jenis entitas. Dapat diamati bahwa model fitur dasar CRF mencapai kinerja terbaik dalam memprediksi entitas P dan CI, dengan *F*1-*score* masing-masing sebesar 0,9854 dan 0,8903. Kinerja terendah dari model fitur dasar CRF dalam memprediksi entitas adalah untuk entitas SC, dengan *F*1-*score* hanya 0,8020.

Kinerja model CRF POS pada tingkat entitas dapat dilihat di Tabel VI. Terlihat bahwa model CRF POS menghasilkan kinerja yang tidak signifikan perbedaannya dari model CRF dengan fitur dasar. Model CRF POS mencapai kinerja yang lebih rendah untuk setiap entitas dibandingkan dengan model CRF dengan fitur dasar.

C. STUDI KASUS: PENCARIAN KUERI PADA LINKED OPEN DATA (LOD)

Setelah membuat *dataset* dan mengembangkan model NER, pencarian dilakukan pada LOD untuk *dataset* statistik. Pencarian dilakukan menggunakan kueri yang ambigu dengan memanfaatkan informasi tentang jenis entitas dan *POS tagging*. Untuk entitas yang dikenali dengan *POS tagging* "NOUN", seperti entitas "populasi", dilakukan pencarian sinonim dalam Tesaurus Bahasa Indonesia untuk mendapatkan tiga sinonim dengan kesamaan tertinggi. Dalam contoh ini, "populasi" memiliki tiga sinonim, yaitu "sampel", "penduduk", dan "individu". Setelah mendapatkan kata sinonim, dilakukan pencarian pada LOD untuk *dataset* statistik menggunakan ekspansi kueri dengan SPARQL. Ekspansi kueri berarti pencarian tidak hanya menggunakan istilah sebagaimana tertulis, tetapi juga memanfaatkan sinonim-sinonim, memungkinkan diperolehnya hasil pencarian bahkan dengan istilah kueri yang ambigu.

Studi kasus pencarian dilakukan pada sepuluh kueri ambigu. Tabel VII menunjukkan perbandingan antara kueri tanpa NER, kueri yang telah melalui NER, dan kueri yang telah melewati NER menggunakan ekspansi kueri sinonim. Berdasarkan Tabel VII, dapat dilihat bahwa tidak semua kueri ambigu dapat

TABEL VII
PERBANDINGAN HASIL PENCARIAN KUERI AMBIGU

No.	Kueri	Hasil Pencarian		
		Tanpa NER	***NER tanpa Ekspansi Kueri Sinonim***	***NER dengan Ekspansi Kueri Sinonim***
1.	populasi jakarta tahun 2019	Tidak ada hasil	Tidak ada hasil	Menemukan hasil dan tepat
2.	data penduduk indonesia 2020	Menemukan hasil dan tepat	Menemukan hasil dan tepat	Menemukan hasil dan tepat
3.	berapa warga jakarta tahun 2019	Tidak ada hasil	Tidak ada hasil	Tidak ada hasil
4.	data masyarakat jawa barat 2019	Tidak ada hasil	Tidak ada hasil	Tidak ada hasil
5.	berapa desa yang ada di aceh 2016	Menemukan hasil dan tepat	Menemukan hasil dan tepat	Menemukan hasil dan tepat
6.	ipm jawa barat 2020	Tidak ada hasil	Tidak ada hasil	Tidak ada hasil
7.	data gender di indonesia tahun 2019	Menemukan hasil dan tepat	Menemukan hasil dan tepat	Menemukan hasil dan tepat
8.	data mortalitas indonesia 2012	Tidak ada hasil	Tidak ada hasil	Menemukan hasil dan tepat
9.	kematian di provinsi banten tahun 2012	Menemukan hasil dan tepat	Menemukan hasil dan tepat	Menemukan hasil dan tepat
10.	data fertilitas indonesia 2012	Tidak ada hasil	Tidak ada hasil	Tidak ada hasil

menemukan hasil pencarian. Kueri ambigu dengan ekspansi kueri pada entitas kueri ambigu tersebut lebih mungkin untuk menemukan hasil pencarian yang akurat dibandingkan dengan kueri tanpa NER dan kueri dengan NER tanpa ekspansi kueri. Kueri tanpa NER dan kueri dengan NER tanpa menggunakan ekspansi kueri memiliki hasil pencarian yang sama, dengan pencarian tanpa NER memerlukan waktu pencarian lebih lama karena melakukan pencarian menggunakan setiap kata, sedangkan kueri dengan NER dapat melakukan pencarian langsung menggunakan entitas yang diidentifikasi.

Beberapa kueri ambigu yang telah diperluas dengan sinonim tidak dapat menemukan hasil pencarian karena istilah dalam sinonim yang diperoleh tidak termasuk dalam istilah dalam LOD. Pada kueri nomor 6, “ipm jawa barat 2020”, tidak ada hasil pencarian yang diperoleh karena tidak ditemukan sinonim untuk “ipm” dalam Tesaurus Bahasa Indonesia, yang seharusnya mengacu pada akronim untuk “Indeks Pembangunan Manusia”. Tesaurus Bahasa Indonesia hanya berisi sinonim untuk akronim yang umum digunakan, tidak mencakup sinonim untuk akronim dalam bidang yang lebih spesifik, seperti statistik demografi sosial.

Pada kueri nomor 3, “berapa warga jakarta tahun 2019” dan kueri nomor 4, “data masyarakat jawa barat 2019”, tidak ditemukan hasil pencarian karena sinonim untuk “warga” dan “masyarakat” yang diperoleh dari Tesaurus Bahasa Indonesia tidak termasuk dalam istilah yang digunakan pada LOD. Dalam LOD, istilah yang digunakan untuk menyatakan ukuran populasi di suatu wilayah adalah “jumlah penduduk”, sementara pencarian sinonim tidak menghasilkan istilah “jumlah penduduk”, sehingga tidak dapat terhubung dengan istilah yang digunakan dalam LOD dan tidak ditemukan hasil pencarian apa pun.

Pada kueri nomor 10, “data fertilitas indonesia 2012”, tidak ditemukan hasil pencarian karena tidak ditemukan sinonim untuk istilah “fertilitas” dalam Tesaurus Bahasa Indonesia, yang seharusnya berkaitan erat dengan istilah “kelahiran” atau “jumlah kelahiran hidup” dalam bidang demografi.

IV. KESIMPULAN DAN SARAN

Pada penelitian ini, telah dibuat sebuah *dataset* dan dilakukan NER pada kueri pencarian *dataset* statistik. Model-model NER telah dikembangkan menggunakan CRF dengan fitur dasar dan CRF dengan fitur dasar ditambah *POS tagging*. Model CRF menggunakan fitur dasar mencapai kinerja NER sebesar 0,9139. Kinerja ini tidak signifikan perbedaannya dari kinerja yang dicapai oleh model CRF dengan POS, yang memiliki *F*1-*score* sebesar 0,9110.

Sebuah studi kasus tentang pencarian sederhana telah dilakukan menggunakan kueri ambigu pada LOD. Hasil pencarian dengan ekspansi kueri yang telah melewati NER dalam kueri menunjukkan kinerja yang lebih baik dalam menemukan hasil pencarian dibandingkan dengan pencarian reguler tanpa NER dan tanpa ekspansi kueri.

Model CRF bergantung pada fitur-fitur yang digunakannya. Fitur-fitur *POS tagging* yang digunakan pada model CRF POS dalam studi ini tidak signifikan meningkatkan model. Untuk penelitian selanjutnya, disarankan untuk mengembangkan lebih lanjut pada *deep learning* yang telah menunjukkan hasil yang menjanjikan dalam tugas NER pada kueri. Selain itu, penelitian lebih lanjut tentang penyelesaian *disambiguation* dapat dilakukan menggunakan *named entity linking* (NEL) dengan menggunakan *dataset* kueri NER yang dibangun untuk menangani ambiguitas pada entitas kueri.

Selain rekomendasi di atas, penelitian selanjutnya juga dapat meningkatkan teknik ekspansi kueri. Hal ini dapat melibatkan pembuatan kamus khusus untuk istilah dalam statistik demografi sosial untuk mengatasi batasan Tesaurus Bahasa Indonesia dalam menemukan sinonim untuk singkatan dan istilah lain yang jarang digunakan. Selain itu, penelitian selanjutnya dapat melakukan ekspansi kueri dengan metode alternatif, misalnya memanfaatkan struktur LOD untuk meningkatkan hasil pencarian.

KONFLIK KEPENTINGAN

Penulis menyatakan bahwa tidak terdapat konflik kepentingan.

KONTRIBUSI PENULIS

Konseptualisasi, Wildannissa Pinasti dan Lya Hulliyyatus Suadaa; metodologi, Wildannissa Pinasti dan Lya Hulliyyatus Suadaa; penulisan—penyusunan draf asli, Wildannissa Pinasti; penulisan—peninjauan dan penyuntingan, Wildannissa Pinasti dan Lya Hulliyyatus Suadaa; validasi, Lya Hulliyyatus Suadaa; supervisi, Lya Hulliyyatus Suadaa.

REFERENSI

[1] R. Song dkk., "Identifying ambiguous queries in web search," dalam *WWW '07*, Proc. *16th Int. Conf. World Wide Web*, 2007, hal. 1169–1170, doi: 10.1145/1242572.1242749.

[2] W. Shen, J. Wang, dan J. Han, "Entity linking with a knowledge base: Issues, techniques, and solutions," *IEEE Trans. Knowl. Data Eng.*, vol. 27, no. 2, hal. 443–460, Feb. 2015, doi: 10.1109/TKDE.2014.2327028.

[3] C. Bizer, T. Heath, dan T. Berners-Lee, "Linked data - The story so far," *Int. J. Semant. Web Inf. Syst. (IJSWIS)*, vol. 5, no. 3, hal. 1–22, 2009, doi: 10.4018/jswis.2009081901.

[4] B.R. Bhange dkk., "Named entity recognition for e-commerce search queries," 2020. Tanggal akses: 30-Jul-2022. [Online]. Tersedia: https://sdm-dsre.github.io/pdf/named_entity.pdf

[5] D. Nadeau dan S. Sekine, "A survey of named entity recognition and classification," *Lingvistic. Investig.*, vol. 30, no. 1, hal. 3–26, Jan. 2017, doi: 10.1075/li.30.1.03nad.

[6] P. Cheng dan K. Erk, "Attending to entities for better text understanding," dalam *Proc. 34th AAAI Conf. Artif. Intell. (AAAI-20)*, 2020, hal. 7554–7561, doi: 10.1609/aaai.v34i05.6254.

[7] D. Mollá, M. van Zaanen, dan D. Smith, "Named entity recognition for question answering," dalam *Proc. 2006 Australas. Lang. Technol. Workshop (ALTW 2006)*, 2006, hal. 51–58.

[8] J. Guo, G. Xu, X. Cheng, dan H. Li, "Named entity recognition in query," dalam *SIGIR '09, Proc. 32nd Int. ACM SIGIR Conf. Res. Dev. Inf. Retr.*, 2009, hal. 267–274, doi: 10.1145/1571941.1571989.

[9] B. Topcu dan I.D. El-Kahlout, "TR-SEQ: Named entity recognition dataset for Turkish search engine queries," dalam *Proc. Recent Adv. Nat. Lang. Process.*, 2021, hal. 1417–1422, doi: 10.26615/978-954-452-072-4_158.

[10] E. Kacprzak dkk., "A query log analysis of dataset search," dalam *17th Int. Conf. ICWE 2017*, J. Cabot, R. De Virgilio, dan R. Torlone, Eds., Cham, Swiss: Springer, 2017, hal. 429–436, doi: 10.1007/978-3-319-60131-1_29.

[11] B. Cowan dkk., "Named entity recognition in travel-related search queries," dalam *Proc. 27th Conf. Innov. Appl. Artif. Intell.*, 2015, hal. 3935–3941, doi: 10.1609/aaai.v29i2.19050.

[12] Y. Wen dkk., "A survey on named entity recognition," dalam *Commun. Signal Process. Syst. (CSPS 2019)*, Q. Liang dkk., Eds., Singapura, Singapura: Springer, 2019, hal. 1803–1810, doi: 10.1007/978-981-13-9409-6_218.

[13] W. Khan dkk., "Named entity recognition using conditional random fields," *Appl. Sci.*, vol. 12, no. 13, hal. 1–18, Jun. 2022, doi: 10.3390/app12136391.

[14] A.S. Wibawa dan A. Purwarianti, "Indonesian named-entity recognition for 15 classes using ensemble supervised learning," *Procedia Comput. Sci.*, vol. 81, hal. 221–228, Mei 2016, doi: 10.1016/j.procs.2016.04.053.

[15] G.B. Herwanto dan D.P. Dewantara, "Traffic condition information extraction from Twitter data," dalam *2018 Int. Conf. Elect. Eng. Inform. (ICELTICs)*, 2018, hal. 95–100, doi: 10.1109/ICELTICS.2018.8548921.

[16] R.M. Yanti, I. Santoso, dan L.H. Suadaa, "Application of named entity recognition via Twitter on spaCy in Indonesian (Case study: Power failure in the Special Region of Yogyakarta)," *Indones. J. Inf. Syst.*, vol. 4, no. 1, hal. 76–86, Agu. 2021, doi: 10.24002/ijis.v4i1.4677.

[17] M.F.D.A. Putra, A.F. Hidayatullah, A.P. Wibowo, dan K.R. Nastiti, "Named entity recognition on tourist destinations reviews in the Indonesian language," *J. Linguist. Komputasional*, vol. 6, no. 1, hal. 30–35, Mar. 2023, doi: 10.26418/jlk.v6i1.89.

[18] Y. Munarko dkk., "Named entity recognition model for Indonesian tweet using CRF classifier," dalam *2017 1st Int. Conf. Eng. Appl. Technol. (ICEAT)*, 2018, hal. 1–6, doi: 10.1088/1757-899X/403/1/012067.

[19] J. Daiber, M. Jakob, C. Hokamp, dan P.N. Mendes, "Improving efficiency and accuracy in multilingual entity extraction," dalam *I-SEMANTICS '13, Proc. 9th Int. Conf. Semant. Syst.*, 2013, hal. 121–124, doi: 10.1145/2506182.2506198.

[20] K. Denistia dan R.H. Baayen, "The morphology of Indonesian: Data and quantitative modeling," dalam *The Routledge Handbook of Asian Linguistics*, 1st ed. Oxfordshire, Inggris: Routledge, 2022.

[21] M. Anandarajan, C. Hill, dan T. Nolan, *Practical Text Analytics: Maximizing the Value of Text Data*, 1st ed. Cham, Swiss: Springer, 2019.

[22] B. Settles, "Biomedical named entity recognition using conditional random fields and rich feature sets," dalam *JNLPBA '04, Proc. Int. Jt. Workshop Nat. Lang. Process. Biomed. Appl.*, 2004, hal. 104–107, doi: 10.3115/1567594.1567618.

[23] M. Poesio dan R. Artstein, "The reliability of anaphoric annotation, reconsidered: Taking ambiguity into account," dalam *Proc. Workshop Front. Corpus Annot. II, Pie Sky*, 2005, hal. 76–83.

[24] R. Rifani, M.A. Bijaksana, dan I. Asror, "Named entity recognition for an Indonesian based language tweet using multinomial naïve Bayes classifier," *Indo-JC (Indones. J. Comput.)*, vol. 4, no. 2, hal. 119–126, Sep. 2019, doi: 10.21108/indojc.2019.4.2.330.

[25] A. Chiche dan B. Yitagesu, "Part of speech tagging: A systematic review of deep learning and machine learning approaches," *J. Big Data*, vol. 9, hal. 1–25, Jan. 2022, doi: 10.1186/s40537-022-00561-y.

[26] J.D. Lafferty, A. McCallum, dan F.C.N. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," dalam *ICML '01, Proc. 18th Int. Conf. Mach. Learn.*, 2001, hal. 282–289.

[27] J. Li, A. Sun, J. Han, dan C. Li, "A survey on deep learning for named entity recognition," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 1, hal. 50–70, Jan. 2022, doi: 10.1109/TKDE.2020.2981314.

[28] B. DuCharme, *Learning SPARQL: Querying and Updating with SPARQL 1.1*, 2nd ed. Sebastopol, CA, AS: O'Reilly Media, 2013.