DOI: 10.22146/jnteti.v13i3.11580

Named Entity Recognition in Statistical Dataset Search Queries

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

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

[Received: 20 January 2024, Revised: 12 April 2024, Accepted: 10 July 2024]
Corresponding Author: Lya Hulliyyatus Suadaa (email: lya@stis.ac.id)

ABSTRACT — Search engines must understand user queries to provide relevant search results. Search engines can enhance their understanding of user intent by employing named entity recognition (NER) to identify the entity in the query. Knowing the types of entities in the query can be the initial step in helping search engines better understand search intent. In this research, a dataset was constructed using search query history from the Statistics Indonesia (Badan Pusat Statistik, BPS) website, and NER in query modeling was employed to extract entities from search queries related to statistical datasets. The research stages included query data collection, query data preprocessing, query data labeling, NER in query modeling, and model evaluation. The conditional random field (CRF) model was employed for NER in query modeling with two scenarios: CRF with basic features and CRF with basic features plus part of speech (POS) features. The CRF model was used due to its well-known effectiveness in natural language processing (NLP), particularly for tasks like NER with sequence labeling. In this research, the basic CRF and the CRF model with POS feature achieved an F1-score of 0.9139 and 0.9110, respectively. A case study on a Linked Open Data (LOD) statistical dataset indicated that searches with synonym query expansion on entities from NER in query produced better search results than regular searches without query expansion. The model's performance incorporating additional POS tagging features did not result in a significant improvement. Therefore, it is recommended that future research will elaborate on deep learning.

KEYWORDS — Named Entity Recognition, Query, Dataset Search, Conditional Random Fields, Linked Open Data.

I. INTRODUCTION

The AllStat is a search application used on the website by Statistics Indonesia (Badan Pusat Statistik, BPS). This search application processes millions of queries to provide statistical data services in BPS, such as accessing datasets, infographics, and statistical publications. One of the fundamental challenges in search engines is understanding ambiguous queries in search to provide relevant search results. Ambiguous queries are queries that have more than one meaning [1] and occur because entities in the queries have many other references, such as aliases, abbreviations, and alternative spellings [2]. For example, the “jumlah penduduk” (population) entity may take other forms like “populasi” (population) or even “jumlah warga” (number of residents) as alternative terms.

According to interviews with subject-matter experts in the BPS, the features of the BPS search engine are expected to search on Linked Open Data (LOD) statistical datasets and better understand user queries, even if the terms used are ambiguous. Linked Data refers to a set of principles for publishing and connecting structured data on the web in a way that machines can read [3]. Linked Data that is published under an open license and collectively referred to is known LOD.

This task can be accomplished using named entity recognition (NER) [4]. NER aims to identify and classify entities from text into predefined categories such as people, locations, and organizations [5]. NER not only acts as a tool for Information Extraction (IE) but also plays a crucial role in various natural language processing (NLP) applications such as text understanding [6], question answering [7], and information retrieval [8].

NER is one of the NLP tasks usually performed on text documents. However, NER models trained on long and grammatically correct sentences often struggle to perform well on queries because queries have different characteristics compared to general text [9]. Queries typically consist of a few words and lack context. A study on log query analysis of dataset searches in a data portal showed that almost 90% of queries consisted of 1 to 3 words, with an average query length of 2.67 [10].

To address this challenge, a study discusses a new problem in A study discussed a new problem in web search called NER in query (NERQ) to address this challenge [8]. This study aimed to detect entities in queries and categorize them into classes. It proposed a probabilistic approach and used the weakly supervised latent Dirichlet allocation (WS-LDA) topic modeling using a specially labeled dataset. The analysis results of this research also showed that around 71% of search queries contained named entities, indicating that identifying named entities in queries can help better understand user search intent.

Another study performed NER on travel-related search queries using machine learning to extract entities from queries [11]. This study applied the conditional random fields (CRF) model with manually labeled datasets, achieving high accuracy results. This research demonstrates that a conventional NLP method is effectively applicable for NER in query in specific domains [11].

Rule-based methods were widely used in the early development of NER. These simple and effective methods depend on establishing a knowledge base and dictionary, resulting in high maintenance costs [12]. While the early NER studies mostly relied on rule-based methods, recent studies have shifted towards supervised learning. Some supervised learning techniques that can be used for NER include hidden Markov models (HMM) and CRF, with CRF algorithms

showing better performance than HMM [5]. CRF is a widely recognized probabilistic model that is useful for various NLP tasks, especially sequence labeling tasks like NER [13].

In Indonesia, NER for the Indonesian language has been extensively developed. One study conducted NER on newspaper articles with 15 entity classes, surpassing the number of classes in existing Indonesian NER [14]. NER has also been applied in various applications using the Indonesian language, such as extracting traffic condition information in the city at a specific given time [15], extracting information related to electrical disturbances [16], and obtaining relevant information for tourists from tourism destination reviews [17]. Another study focused on NER in shorter texts, such as NER in Indonesian tweets using CRF [18]. However, there has been no research on NER in Indonesian queries. It has been acknowledged that language-aware methods in NLP are more accurate than language-independent methods [19]. Specifically, the Indonesian language has rich derivational morphology, including reduplication [20]. Capturing the morphological through language-specific datasets can improve the performance of NER, including NER in Indonesian queries.

Based on the background, this research aimed to model NER in query to extract entities in statistical search queries. The contributions are summarized as follows.

1. Constructing a dataset for NER in statistical dataset-related search queries using search history data from the BPS website.
2. Developing NER in query models with CRF.
3. Conducting a case study of search on LOD containing BPS statistical dataset data using queries that have identified their entity types through NER in query.

II. METHODOLOGY

This research focused on analyzing queries related to searching statistical datasets in socio-demographics statistics from 2020 to 2022. The stages in this research included query data collection, query data preprocessing, labeling query data to form a dataset, NER in query model development, and model evaluation. The research flow can be seen in Figure 1.

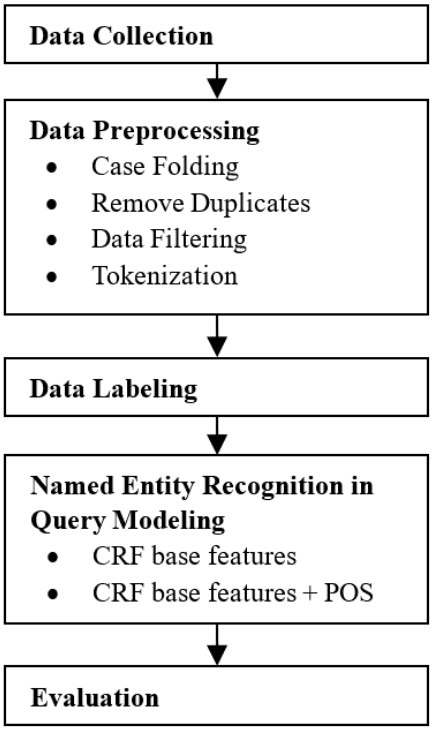

Figure 1. Research flow.

A. DATA COLLECTION

The data used in this research were user search queries from the AllStat application used on BPS' website. The collected data included query data from 2020 to 2022 with attributes such as query, year, total monthly searches, and total searches within a year.

B. DATA PREPROCESSING

Data preprocessing is the process of preparing the collected data for further processing or analysis in the subsequent stages [21]. In preprocessing, there are several stages.

1) CASE FOLDING

Case folding aims to convert all letters in the queries to lowercase. In the obtained query data, queries are recorded with case-sensitive rules, hence it is necessary to perform case folding to lowercase to standardize text representation so that there is no difference between uppercase and lowercase letters in text processing.

2) REMOVE DUPLICATES

Removing duplicates is the process of identifying and removing identical queries, resulting in unique ones. In the obtained query data, queries and their frequencies were recorded based on the year, allowing for the possibility of having the same query in different years. Additionally, case folding results in the previous stage enabled queries that were initially recorded differently to become the same due to their conversion to lowercase. Therefore, duplicates are removed to obtain unique queries.

3) DATA FILTERING

The data filtering stage eliminated irrelevant queries not related to searching statistical datasets in socio-demographic statistics. This process ensured that only relevant queries were retained for subsequent stages of analysis. Query data that underwent previous preprocessing stages were semi-automatically filtered during data filtering. Automated filtering was achieved by establishing rules, such as compiling a list of irrelevant words commonly found in irrelevant queries, to facilitate the removal of queries containing these words.

4) TOKENIZATION

Before labeling, the filtered query data were tokenized. Tokenization is the process of dividing sentences into tokens or specific parts.

C. DATA LABELING

As done in several previous studies [11], [22], the entity classes for labeling can be adjusted according to the domain. The label classes in this research were determined based on the statistical domain, adapting the concepts used in the BPS. The entity class labels used in this research include:

1) STATISTICAL INDICATOR (SI)

Statistical indicator (SI) describes the characteristics of economic, social, and other phenomena at a specific time and place. Examples: "jumlah penduduk" (total population).

2) STATISTICAL CLASSIFICATION (SC)

A set of categories that represent the values of one or more variables listed in statistical surveys or administrative data, used in the production and dissemination of statistics. Examples: "jenis kelamin" (sex), "provinsi" (province).

3) CLASSIFICATION ITEM (CI)

A set of categories at a certain level in the statistical classification that define the content and boundaries of those categories, for examples: “perempuan” (female), “dki Jakarta.”

4) PERIOD (P)

The period represents the time of the statistical data user wants to search, for example: 2020. The labeling process used the beginning, inside, and outside (BIO) notation. For each identified entity belonging to an entity class X, the tokens at the beginning of the entity were labeled as B-X. If the recognized entity consisted of two or more tokens, the second and subsequent tokens within that entity were labeled as I-X. Tokens that did not belong to any entity class were labeled as O.

The labeling process is semi-automated. Automated labeling used several routes to expedite the labeling process and reduce the burden on human annotators. Manual labeling was performed on tokens that had not been labeled using automated labeling. In addition, it was done to correct the results of automated labeling. Automatic labeling was carried out using several rules as follows.

1. Years are classified as period entity class with the label P. Labeling years as period was done on tokens consisting of four digits using the assistance of RegEx.
2. Regional names were classified as CI entity class with the label CI. This labeling used a list of province and district/city names in Indonesia, along with synonyms and abbreviations. Tokens in the query were labeled as CI if they matched the word list.
3. Gender was classified as CI entity class with the label CI. This labeling used a list of genders along with their synonyms.
4. Names of statistical indicators were classified as SI entity class with the label SI. In this labeling, a list of statistical indicators in the field of social demographics from BPS is used.
5. Stop words such as connecting words like “menurut” (according to) was not classified into any entity class and were therefore labeled as outside with the label O. This labeling used a list of stop words in Indonesian from the Python Sastrawi library.

Two annotators equipped with labeling guidelines manually labeled the remaining unlabeled tokens. Inter-rater reliability was calculated using Krippendorff’s alpha method to measure the quality of query labeling [23].

In addition to entity class labeling for NER, part-of-speech (POS) tagging was also performed to enhance the model performance, as demonstrated in several prior research efforts in general NER using CRF [24]. POS tagging is a process aimed at categorizing words into classes based on their positions in a sentence [25]. These word classes can include verbs, adjectives, and nouns. The results of POS tagging were used as additional features to improve the performance of the CRF NER. The Flair library in Python was used for POS tagging.

D. NER IN QUERY MODELING

NER in query modeling in this research was done using the linear chain CRF model to identify entity types in queries [26]. CRF is a widely recognized probabilistic model that is useful for various NLP tasks, especially sequence labeling tasks like NER [13].

CRF belongs to the category of undirected graphical models that can be used to compute the conditional probability of an output sequence $y = <y_1, \ldots, y_T>$ given an input sequence $x = <x_1, \ldots, x_T>$. Z is the normalization factor for state sequences. In NER, the input x is a sequence of tokens in a query, while the output y is a sequence label representing the segmentation of codes and entity class information (e.g., B-SI to represent the entity prefix for SI). λk is the value of the weight learned for the feature 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. EVALUATION

The NER model is evaluated on the test data by comparing the output with human labeling as measured by precision, recall, and F1-score calculations [27]. Precision, recall, and F1-score are computed based on the counts of true positive (TP), false positive (FP), and false negative (FN).

1. TP occurs when the entity recognized by NER matches the ground truth.
2. FP occurs when the entity recognized by NER does not match the ground truth.
3. FN occurs when the entity labeled in the ground truth is not recognized by NER.

The formulas for precision, recall, and F1-score are as follows.

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

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

$$F_1 - score = \frac{Precision .Recall}{Precision + Recall}. \quad (4)$$

F. CASE STUDY: QUERY SEARCH ON LINKED OPEN DATA (LOD)

After constructing the dataset and performing NER in query modeling, a case study of search queries that passed the NER stage was conducted to search on the LOD BPS statistical dataset.

The purpose of this case study was to provide an overview of the usage of queries that underwent NER to help provide context. In this research, searches were attempted on LOD, especially for queries with ambiguous terms.

However, not all entities in the queries identified by the NER process used the standard terms used by BPS. For example, BPS uses the term “jumlah penduduk” (population) as a SI for data describing the population count in a region. Still, there are queries that used the term “populasi” (population) to inquire about the same data. On the other hand, LOD was built with standard entities referring to the metadata management system (MMS) at BPS. The difference in terms between user-mentioned entities and those present in LOD necessitates a process to connect these entities to achieve search results that align with the user’s intent.

This research conducted query expansion to address this issue by searching for synonyms for ambiguous entities so that the search included not only the mentioned terms but also their synonyms. The search for synonym words was performed using the Indonesian Thesaurus. This study selected three synonyms with the highest similarity for query expansion. Subsequently, LOD data searches were conducted using SPARQL protocol and RDF query language (SPARQL). SPARQL is a query language and standard protocol for LOD on the web or for data with the RDF model [28]. The case study of searches on LOD was conducted by comparing the query

Jurnal Nasional Teknik Elektro dan Teknologi Informasi
Volume 13 Number 3 August 2024

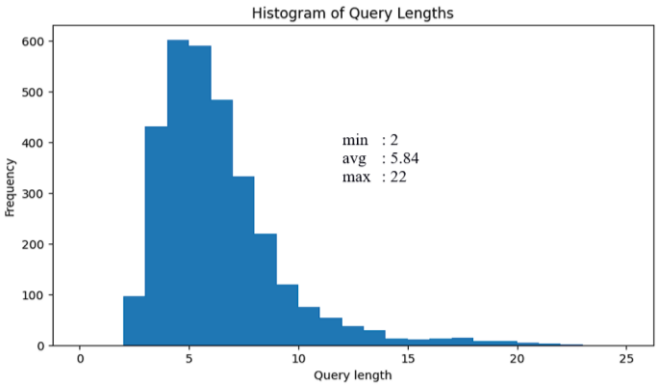

Figure 2. Histogram of query lengths.

TABLE I
LABELING RESULTS AT THE TOKEN LEVEL

Token Label	Number of Tokens
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
Total	18,196

searches with synonym expansion and the original queries without synonym expansion. Further, a human evaluation was conducted to compare the two search results.

III. RESULTS AND DISCUSSIONS

A. DATASET

The data obtained for this research consisted of 2,324,645 queries. However, these queries must go through a preprocessing stage to be prepared for labeling. The preprocessing process began with converting all words in the queries to lowercase to standardize query representation. After that, duplicates were removed, and semiautomatic filtering was performed to obtain relevant queries, resulting in 3,145 queries. These queries were then tokenized, producing 18,196 tokens ready for labeling.

Figure 2 illustrates the distribution of query lengths that is used to create the dataset. The average query length was 5 to 6 words, with the shortest query consisting of two words and the longest query found in queries with 22 words.

Labeling was done semiautomatically on the 18,196 tokens. The total number of tokens labeled automatically was 12,372 tokens, accounting for 68.04% of the total tokens. Then, manual labeling was performed by two annotators, including checking the results of automatic labeling. In this study, an alpha value of 0.6515 was obtained, indicating a sufficient level of annotator agreement. The distribution of labeling results at the token level can be seen in Table I and the distribution at the entity level can be seen in Table II.

Examples of the query that has been labeled can be seen in Table III. In the example, the phrase "jumlah penduduk" (total population) in the query was identified as an entity belonging to the SI class. Therefore, the word "jumlah" (total) was labeled as B-SI (begin) indicating the beginning of a SI entity, while the subsequent word "penduduk" (population) was labeled as

TABLE II
LABELING RESULTS AT THE ENTITY LEVEL

Token Label	Number of Tokens
SI	3,219
SC	505
CI	1,921
P	3,070
O	3,054
Total	11,769

TABLE III
EXAMPLE OF LABELED DATA

Query	Token	Label
"jumlah penduduk dki jakarta menurut kelompok umur tahun 2020" (total population of dki jakarta by age group in 2020)	"jumlah" (total)	B-SI
	"penduduk" (population)	I-SI
	"dki" (dki)	B-CI
	"Jakarta" (jakarta)	I-CI
	"menurut" (by)	O
	"kelompok" (group)	B-SC
	"umur" (age)	I-SC
	"tahun" (year)	O
	2020	B-P

	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

Figure 3. Performance of POS tagging.

I-SI (inside) to indicate that it was inside the SI entity. The same labeling pattern applied to other recognized entities, where each recognized entity began with a begin label, and any subsequent words within the same entity were labeled as inside. Outside was used to label words that did not belong to any entity class.

In addition to entity labeling using NER, this study also performed POS tagging as an additional feature in the model. The Flair library in Python was used for POS tagging. The model was trained using a provided corpus by Flair. The training was conducted in Google Collaboratory, and the model achieved an F1-score of 93.59% on the test data. Figure 3 displays the model performance in recognizing each class. The model could predict SYM and PUNCT with the best performance among all POS tagging, with an F1-score being 100 % and 99.85 %, respectively. Nevertheless, the model failed to predict the X tag, as the performance generated for this tag was 0%. Low prediction performance was also shown for

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

Figure 4. Snapshot of dataset.

the ADJ tag, which achieved a performance of 82.79%. The model was then used to predict POS tags for all tokens. A snapshot of the constructed dataset is provided in Figure 4. The figure shows several rows from the dataset, presenting information about "query," "token," "postage," and "label."

B. NER IN QUERY MODELING AND EVALUATION

NER modeling with CRF was done using the Python CRFsuite library. Two scenarios were employed: CRF with basic features and CRF with basic features and POS (after this, also referred to as CRF POS). The basic features used in modeling were at the word level, including the word itself, suffix features, and whether the word form is a digit. This study did not use case features since the data were already lowercase.

Each CRF model scenario used hyperparameters based on the grid search results. Table IV shows the experimental results carried out using 5-fold cross-validation. Based on the performance results, the CRF model with basic features outperformed the CRF model with additional POS features, although the difference in F1-score was not significant. The F1-scores for CRF with basic features and CRF POS were 0.9139 and 0.9110, respectively. The low performance of the CRF model with POS features may be attributed to the short and incomplete nature of the queries. In addition, 60% of the POS tagging labels in the dataset are NOUN, which did not significantly contribute to improving the learning of the NER in the query model.

In addition to overall model performance, the performance of the NER model could also be examined at the entity level. Table V displays the performance of the CRF base feature model for each entity type. It can be observed that the CRF base feature model achieved the best performance in predicting P and CI entities, with F1-scores of 0.9854 and 0.8903, respectively. The CRF basic feature model had the lowest performance in predicting SC entities, with an F1-score of only 0.8020.

The performance of the CRF POS model at the entity level can be seen in Table VI. It can be seen that the CRF POS model produced performances that were not significantly different from the base feature CRF model. The CRF POS model achieved lower performance for each entity compared to the CRF base feature model.

C. CASE STUDY: QUERY SEARCH ON LINKED OPEN DATA (LOD)

After creating the dataset and developing the NER model, searches were conducted on the LOD statistical dataset. The search was performed using ambiguous queries by leveraging

TABLE IV
PERFORMANCE OF THE NER QUERY

Model	Precision	Recall	F1-Score
CRF-base feature	0.9249	0.9031	0.9139
CRF POS	0.9215	0.8999	0.9110

TABLE V
PERFORMANCE OF THE CRF BASE FEATURE MODEL

Entity	Precision	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

TABLE VI
PERFORMANCE OF THE CRF BASE FEATURE + POS MODEL

Entity	Precision	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

the information about the entity types and their POS tagging. For entities recognized with NOUN POS tagging, such as the entity "populasi" (population), a search for synonyms was conducted in the Indonesian Thesaurus to obtain three synonyms with the highest similarity. In this example, "populasi" has three synonyms: "sampel" (sample), "penduduk" (residents), and "individu" (individual). After obtaining synonym words, a search was conducted on the LOD statistical dataset using query expansion with the SPARQL. Query expansion means the search not only uses the terms as written but also leverages their synonyms, allowing for search results even with ambiguous query terms.

A case study of searches was conducted on 10 ambiguous queries. Table VII shows the comparison between the search results of queries without NER, queries that have undergone NER, and queries that have undergone NER and expanded with query expansion using synonyms. Based on Table VII, it can be seen that not all ambiguous queries could find search results. Ambiguous queries with NER using query expansion on their entities were more likely to find accurate search results compared to queries without NER and query with NER using query expansion. Queries without NER and queries with NER using query expansion had the same search results. Searches without NER required more time as they used each word, while queries with NER could perform direct searches using the identified entities.

Some ambiguous queries that had expanded queries with synonyms could not find search results because the terms in the synonyms obtained were not included in the terms in the LOD. For query number 6, no search results were obtained for "ipm jawa barat 2020" (west java hdi 2020) because no synonym for 'ipm' (hdi) was found in the Indonesian Thesaurus, which should refer to the acronym for *indeks pembangunan manusia* (human development index). The Indonesian Thesaurus only contains synonyms for acronyms that are generally used but does not include synonyms for acronyms in more specific fields such as social demographic statistics.

There were nor results found for query number 3 "berapa warga jakarta tahun 2019" (how many jakarta residents in 2019) and query number 4 "data masyarakat jawa barat 2019" (west java residents 2019) because the synonyms for "residents" and

TABLE VII
COMPARISON OF AMBIGUOUS QUERY SEARCH RESULTS

No	Query	Search Results		
		Without NER	***NER Without Synonym Query Expansion***	***NER With Synonym Query Expansion***
1.	"populasi jakarta tahun 2019" (jakarta population in 2019)	No results	No results	Found accurate results
2.	"data penduduk indonesia 2020" (indonesia population data 2020)	Found accurate results	Found accurate results	Found accurate results
3.	"berapa warga jakarta tahun 2019" (how many jakarta residents in 2019)	No results	No results	No results
4.	"data masyarakat jawa barat 2019" (west java residents 2019)	No results	No results	No results
5.	"berapa desa yang ada di aceh 2016" (how many villages are there in aceh 2016)	Found accurate results	Found accurate results	Found accurate results
6.	"ipm jawa barat 2020" (west java HDI 2020)	No results	No results	No results
7.	"data gender di indonesia tahun 2019" (gender data in Indonesia in 2019)	Found accurate results	Found accurate results	Found accurate results
8.	"data mortalitas indonesia 2012" (indonesia mortality data 2012)	No results	No results	Found accurate results
9.	"kematian di provinsi banten tahun 2012" (deaths in the province of banten in 2012)	Found accurate results	Found accurate results	Found accurate results
10.	"data fertilitas indonesia 2012" (indonesia fertility data 2012)	No results	No results	No results

"community" obtained from the Indonesian Thesaurus were not included in the terms used in the LOD. In the LOD, the term used to express the population size in an area is "jumlah penduduk" (population), while the search for synonyms did not retrieve this term. Hence, it could not connect with the terms used in the LOD and could not find any search results.

For query number 10, no search results were found for "data fertilitas indonesia 2012" (indonesia fertility data 2012) because there is no synonym for the term "fertilitas" (fertility) found in the Indonesian Thesaurus, which should be closely related to the terms "birth" or "number of live births" in the field of demography.

IV. CONCLUSION AND FUTURE WORK

The researchers have constructed a dataset and performed NER on statistical dataset search queries. NER models were developed using CRF with basic features and CRF with basic features plus POS tagging. The CRF model using basic features achieved an NER performance of 0.9139. This performance is not significantly different from that achieved by the CRF model with POS, which had an F1-score of 0.9110.

A case study was conducted using a simple search using ambiguous queries on the LOD. The search results with query expansion on entities from NER in the query process demonstrated better performance in finding search results than regular searches without query expansion.

The CRF model depends on the features it uses. The POS tagging features used in the CRF POS model in this study could not significantly enhance the model. For future research, it is suggested that deep learning be elaborated on, as it has shown promising results in NER query tasks. Additionally, further research on addressing disambiguation can be conducted using named entity linking (NEL) and the NER query dataset constructed to handle ambiguity in query entities.

In addition to the recommendations mentioned above, future research can also enhance query expansion techniques. It can involve the creation of a specialized dictionary for terms in social demographic statistics, addressing the limitations of the Indonesian Thesaurus in finding synonyms for abbreviations and other infrequently used terms. Furthermore, conducting query expansion through alternative methods, such as harnessing the LOD structure, presents an opportunity to bolster search results.

CONFLICTS OF INTEREST

The authors declare no conflicts of interest.

AUTHORS' CONTRIBUTIONS

Conceptualization, Wildannissa Pinasti and Lya Hulliyyatus Suadaa; methodology, Wildannissa Pinasti and Lya Hulliyyatus Suadaa; writing – original draf preparation, Wildannissa Pinasti; writing – review & editing, Wildannissa Pinasti and Lya Hulliyyatus Suadaa; validation, Lya Hulliyyatus Suadaa; supervision, Lya Hulliyyatus Suadaa.

REFERENCES

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

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

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

[4] B.R. Bhange *et al.*, "Named entity recognition for e-commerce search queries," 2020. Access date: 30-Jul-2022. [Online]. Available: https://sdm-dsre.github.io/pdf/named_entity.pdf

[5] D. Nadeau and S. Sekine, "A survey of named entity recognition and classification," *Lingvistic. Investig.*, vol. 30, no. 1, pp. 3–26, Jan. 2007, doi: 10.1075/li.30.1.03nad.

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

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

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

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

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

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

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

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

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

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

[16] R.M. Yanti, I. Santoso, and 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, pp. 76–86, Aug. 2021, doi: 10.24002/ijis.v4i1.4677.

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

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

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

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

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

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

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

[24] R. Rifani, M.A. Bijaksana, and 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, pp. 119–126, Sep. 2019, doi: 10.21108/indojc.2019.4.2.330.

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

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

[27] J. Li, A. Sun, J. Han, and C. Li, "A survey on deep learning for named entity recognition," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 1, pp. 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, USA: O'Reilly Media, 2013.