International Journal on Advanced Science Engineering Information Technology

Vol.14 (2024) No. 5
ISSN: 2088-5334

Automatic Semantic Annotation of Indonesian Language Phrase Using N-Gram Language Model

Dewi Wardani [a,*], Chania Evangelista [b]

[a] Department of Data Science, Universitas Sebelas Maret, Jl Ir Sutami 36 A Surakarta, Indonesia
[b] Department of Informatics, Universitas Sebelas Maret, Jl Ir Sutami 36 A Surakarta, Indonesia
Corresponding author: *dww_ok@uns.ac.id

Abstract— **Building semantic data populations in unstructured data or text is challenging. In this type of data, several problems can be raised, some of which are difficult to analyze. Some groups of words or expressions cannot be defined according to their meaning and can be a source of ambiguity. It can have a different meaning depending on the context of its use. This work aims to automatically annotate Indonesian Language text, especially phrases, with the existing knowledge base. The result is text with semantic markup. Machines can automatically process this type of text because it describes its meaning. This work applies an n-gram language model to identify meaningful phrases and defines them as a unit so that every existing word or phrase is automatically semantically tagged. This work uses the DBpedia and *schema.org* knowledge base. The percentage of successfully labeled data in this job was 78% with 84.95% accuracy using DBpedia and 5.9% with 97.46% accuracy using schema .org. Some factors affect the accuracy score, including the availability of the required data with the data contained in the knowledge base, the system's ability in the POS tagging process, and many new terminology and local cultures that have not yet been contained in the knowledge bases, especially *schema.org* that is utilized as a standard for all search engines. This work will help the machine understand the semantics of text data. All pages obtained will be semantically tagged and, therefore, will be understood by machines. This ability will support the following processes.**

Keywords— **Automatic semantic tagging; data integration; DBpedia; N-gram; *schema.org***

Manuscript received 10 Sep. 2023; revised 11 May 2024; accepted 4 Jun. 2024. Date of publication 31 Oct. 2024.

I. INTRODUCTION

Data in text form can be divided into three categories: structured, semi-structured, and unstructured [1]. The population of documents on the internet is vast and often used, but documents with semantic tagging still need to be made available, especially in low-resource language. Semantic tagging and semantic annotation [2] are part of implementing Semantic web technologies [3], and semantic web [4] aims to make data understand its meaning. Therefore, machines can process automatically. The first step is semantic tagging to enrich the existing text with a distinct meaning [5]. Documents are challenging to be analyzed [6]. Therefore, several approaches have been proposed, including [7], [8], [9], [10], [11]. Giving meaning can make accessing content more accessible [12]. More of the population of Semantic Web data needs to be structured, mainly in low-resource languages like Indonesian. Search engines also better index documents with semantic tags. Search engines use *schema.org* as a unified lexical ontology to mark words as microdata markup. This will make it easier for search engines to process the [13] document.

Semantic markup is necessary because the meaning contained in the data enriches the document. Semantic tagging increases the efficiency of text analysis, so this technology has been applied in various fields [14], [15], [16]. For example, they extract documents to create more structured data, generate precise metadata and business analysis, and analyze articles to retrieve them. Decision-making, social media monitoring, reputation management, and contextual advertising enable better placement of ads on the site so that advertising objectives can be delivered accurately. Until now, documents have only been parsed using syntax-based natural language processing (NLP) technology, ignoring the meaning of document enrichment.

One challenge in document semantic annotation is that ambiguity can lead to misinterpretation of information. Another problem is annotating phrases (sets of words) with a single, inseparable meaning. An example is a phrase in Indonesian Language, *"Hati hati"* semantically means

"careful", but every word *"hati"* means *"heart"*. Therefore, a technology capable of enriching the text with an accurate understanding of the data is essential. This technology enables meaning by defining each word and phrase as a unit according to an internationally recognized text. One method that can be used is the n-gram language model, which has the advantage of simplicity and scalability in handling data more efficiently. By predicting the next word in a phrase, the n-gram language model can be used effectively and comprehensively to capture the regularity [17]. N-grams have been successfully deployed on topics in big data warehouses or other extraction processes, but not semantic annotation [18].

Structured data, as semantic markup for textual data, includes several properties and confidence that contain integers from 0 to 100 that indicate the semantic analyzer's confidence in the expression. The Solution, the grammar is the grammar rules that correspond to the interpretation, and the x model is the location of the XForms model data used to interpret [19]. Semantic technology knowledge bases, often called ontology, can give meaning to documents because semantic technology provides a unit of meaning. Semantic technology is the most common technique for finding the meaning of a document [20]. By providing an understanding of information, semantic technology can refer to a standardized knowledge base (KB) that all users or machines can understand. Knowledge bases gather much factual knowledge, and their scope and completeness vary widely between different[21] domain types. Web semantic annotation is a way to generate high-quality [22] content or data. Semantic annotation to prepare data for automated processing is part of the goal of the underlying semantic technology [4]. From a semantic web perspective, semantic resources must be linked to existing knowledge bases or ontology [23]. A work using Indonesian Language phrases gives another direction.

This work uses phrases to extract a single keyword [24]. Up to now, work on semantic annotation for Indonesian Language documents is still being carried out. This work proposes a new point in determining a uniquely meaningful Indonesian Language phrase using the n-gram language modeling method and automatically providing semantic labels. One of the goals of delivering semantic markup is to enrich the Semantic Web's data in the Indonesian Language. These two goals are the contribution of this work. It is rare to find semantic tags in the Indonesian Language. In this work, the annotation process to generate semantic tagging can be done automatically using the Indonesian Language corpus. This system uses a standardized knowledge base to provide rich semantic tagging. This work resulted in the markup of information in the form of Extensible Markup Language (XML) with knowledge base sources from DBpedia and microdata with *schema.o*rg sources to provide information that all users and all machines can understand. Significant work has yet to be done on semantic annotation in Indonesian.

II. Materials and Method

A. Phrase Recognition Using N-Gram Language Model

Santos et al. [17] concluded that using the n-gram method is suitable for studying patterns and efficiently capturing the rules in the API. Other related work studies concluded with similar arguments. These studies have shown positive results using the N-gram language model [25], [26], [27]. This step is intended to identify existing data as a word or a meaningful phrase unit. In this work, N-grams used the fourth edition corpus of the Large Indonesian Language Dictionary (KBBI) and predicted the possible words following each word. This work uses n-grams of length two (bigram) and three (trigram). In an n-gram of length 2, each word is processed to obtain the next prediction and generate a probability with the formula derived from formula one as follows:

$$P(w_n|w_{n-1}) = \frac{C(W_n|W_{n-1})}{C(W_{n-1})} \quad (1)$$

Where P is the probability or likelihood, w_n is the prediction of the word after that word is the followed word, and C is the count, i.e., a number. The probability of occurrence of each word and the word that follows it in the data is calculated and then divided by the number of occurrences tracked to generate the probability. In an n-gram of length three, every two words are processed to derive the next prediction and generate the probability with the formula derived from formula two as follows:

$$P(w_n|w_{n-2,}w_{n-1}) = \frac{C(W_{n-1}|W_{n-2})}{C(w_{n-2},w_{n-1})} \quad (2)$$

Algorithms 1 and 2 show a part of the implementation, as shown in Figures 1 and 2.

Algorithm 1: Implementation of n-gram

Input: KBBI
Output: w1, w2, w3, probability

```
model = defaultdict(Λ: defaultdict(Λ: 0))
for sentence ∈ KBBI do
    for w1, w2, w3 ∈ trigram(sentence) do
        model[(w1, w2)][w3] += 1
for w1, w2 ∈ model do
    totalcount = float(∑(model[w1_w2].values()))
    for w3 ∈ model[w1_w2] do
        model[w1_w2][w3] = totalcount
model2 = defaultdict(Λ: defaultdict(Λ: 0))
for sentence ∈ KBBI do
    for w1, w2 ∈ bigram(sentence) do
        model[(w1)][w2] += 1
for w1 ∈ model2 do
    totalcount = ∑(model2[w1].values())
    for w2 ∈ model[w1] do
        model2[w1][w2] = totalcount
```

Fig. 1 Algorithm 1, an Algorithm of Implementation of n-gram

Suppose there is a word, and the prediction of the word after entering that word is the same as the previous prediction in the corpus training data with a probability value greater than 0.003. In this case, the word group is identified as a one-way phrase. The number 0.003 is derived from researchers' observations and tests of several possible phrases.

Algorithm 2: Algorithm for Obtain Unigram, Bigram, Trigram

```
Input: InputText
Output: TokenizedText
Trigram(InputText,TokenizedText)
tokens = tokenize(InputText)
for n ∈ tokens do
    for i ∈ range(len(tokens)-2) do
        if tokens[i] = n then
            n2 = tokens[i+1]
            n3 = tokens[i+2]
            ngram = dict(model[n, n2])
            for x, y ∈ ngram.items() do
                if x=n3 ∧ y > 0.8 then
                    result = n, n2, n3
                    for a ∈ result do
                        resulttrigram = tokenizer.add_mwe([n, n2, n3])
resulttri = trigram(InputText)
tokenfrasa = tokenize(InputText.split())
Bigram(InputText,TokenizedText)
for n ∈ tokenfrasa do
    ngram = dict(model2[n])
    for i ∈ range((tokenfrasa)-1) do
        if tokenfrasa[i] = n then
            result = tokenfrasa[i+1]
            for x, y ∈ ngram.items() do
                if (x=result ∧ y > 0.003) then
                    result = n, result
                    for a ∈ result do
                        resultbigram = tokenizer.add_mwe([n, result])
resultbi = bigram(tokenfrasa)
tokenfrasa2 = tokenizer.tokenize(InputText.split())
```

Fig. 2 Algorithm 2, Algorithm to obtain Unigram, Bigram, Trigram

Sample results of the N-Gram implementation from the sample text are shown in Table 1. The main objective is to obtain phrases that are at least in bigram but will also obtain unigrams. Find expressions important for semantic markup (next step). For example, the phrase "*dewan perwakilan rakyat*" bigram phrase has only one semantic, parliament, instead of three semantic words, "*dewan*" (council), "*perwakilan*" (representative), *"rakyat"* (person).

B. Semantic Tagging

First, the POS tagging process needs to be done. It provides categories for each word and phrase that affect the tag identification step. The Indonesian language has its own POS categories. There are 21 categories; CC (coordinating conjunction), CD (main number), OD (ordinal number), DT (qualifier), FW (a foreign word), IN (preposition), JJ (adjective, MD (modal), and auxiliary verbs), NEG (negative), NN (noun), NNP (proper noun), NND (measurable noun), PR (indicative pronoun), PRP (personal pronoun), RB (adverb), RP (grain), SC (subordinate conjunction)), Sym (Symbol), VB (Verb), WH (Interrogative word) and Z (Punctuation) Table 1 shows examples about the POS obtained from the dataset. Meanwhile, Table 4 shows the details of POS in the Indonesian language.

The POS tagging step identifies and tags a word or phrase based on its type according to the token distribution or the prefix of the previous [28] word. The POS tagging process uses a data store based on data from Fam Rashel, which has around 200,000 tokens. This step is necessary to define a markup-related word in DBpedia and *schema.org*. There are two sections in DBpedia, namely resources and properties. A resource is a web page containing a more detailed explanation of a tag word or phrase. At the same time, the property refers to a word or phrase with a verbal label. In *schema.org*, verbs are defined in the class "Action", and nouns or words are defined in "Things".

TABLE I
THE EXAMPLE RESULT OF N-GRAM AND ITS OBTAINED POSTAG

No	Word (in Indonesian Language	Form	POSTag
1	Gubernur	Unigram	NN
2	Anggota Dewan	Bigram	NN
3	Dewan Perwakilan Rakyat	Trigram	NN

The next step is to provide tags for each term. The final markup step is to run a query that finds each desired piece of data in the DBpedia knowledge base. This process uses the DBpedia endpoint. This job uses SPARQL with select queries. Besides DBpedia, schema.org is used to generate microdata. Search engines use microdata to create semantic indexes. As a result, text with semantic tagging will be indexed and made more accessible by search engines. A translation process is required because all semantic markup terms are presented in English. DBpedia will provide the results as XML to the system if the requested data is found. The XML data is then combined with the HTML to form a web page containing all the words entered by the user. Besides DBpedia, it also uses *schema.org*. Each text data will be labeled as an Action or Article class. The resulting microdata will be combined with HTML to create a web page. For words without tags, it just shows text without links. Meanwhile, tagged data will have a link that directs the user to the *schema.org* page explaining the word.

If the resulting terms are labeled, then there are two SPARQL models to obtain resources that tend to be of type NN and properties that tend to be of type VB. This work puts the limits of NN and VB first. In Indonesian, NN is the subject or object, and VB is the predicate. Table 2 shows the SPARQL template to get the markup reference from DBpedia and *schema.org* in Table 3.

TABLE II
THE TEMPLATE OF SPARQL ON DBPEDIA FOR OBTAINING TAGGING

Type	SPARQL
Subject	select distinct ?tag where { ?tag ?b ?c . ?tag ?d dbc:Resources .}
Predicate	select distinct ?tag where { ?a ?b ?tag . ?tag ?d dbc:Resources .}
Object	select distinct ?tag where { ?a ?tag ?c . ?tag ?d rdf:Property . }

TABLE III
THE TEMPLATE OF SPARQL ON *SCHEMA.ORG* FOR OBTAINING TAGGING

Type	SPARQL
Subject	select distinct ?tag where {?tag ?b ?c . ?tag ?d rdf:Class .}
Predicate	select distinct ?tag where {?a ?b ?tag . ?tag ?d rdf:Class .}
Object	select distinct ?tag where {?a ?tag ?c . ?tag ?d Class:Action .}

TABLE IV
THE TEMPLATE OF SPARQL ON *SCHEMA.ORG* FOR OBTAINING TAGGING

No	POS	Explanation	Example (in the Indonesian Language)
1	CC	Coordinating conjunctions are words connecting two words, phrases, or clauses that have an equal position in the sentence structure	*dan, tetapi, atau*
2	CD	A cardinal number is a word that denotes a number or the number itself	*dua, lima, ribuan, ke-4, 2020, 253*
3	OD	Ordinal numbers are words that indicate stage	*pertama, kedua, ketiga*
4	DT	A determiner or clause is a word that limits nouns	*para, sang, si*
5	FW	Foreign words are words in foreign languages that have yet to be absorbed into Indonesian	*read, change, term*
6	IN	Prepositions	*dalam, di, ke, oleh, pada*
7	JJ	The adjective	*bersih, panjang, lama*
8	MD	Modal and auxiliary verbs are words that function to help verbs	*boleh, harus, sudah, mesti, perlu*
9	NEG	Negation is a word that denotes a negative statement	*tidak, belum, jangan*
10	NN	Nouns refer to humans, animals, objects, concepts, or meanings	*baju, meja, singa, tangan*
11	NNP	A proper noun is the specific name of a person, thing, or place	*Boediono, Indonesia, Bank Mandiri, Januari*
12	NND	Measurement nouns are words that place a noun into a certain number of groups. Size nouns refer to size, distance, volume, speed, weight, or temperature	*halaman, ton, menit, buah*
13	PR	Demonstrative pronouns are pronominal pointers to something	*itu, ini*
14	PRP	The personal pronoun is the pronoun used to refer to people	*saya, kami, kita, dia, mereka*
15	RB	Adverbs	*sangat, hanya, segera*
16	RP	A particle is a word that has no lexical meaning but can have meaning when combined with other words	*pun, -lah, -kah*
17	SC	A subordinating conjunction is a word that connects two or more clauses, and one of these clauses is a subordinate clause	*jika, meski, maka, dengan, bahwa, yang, semoga*
18	SYM	Symbols given to POS tagging include mathematical symbols and currency symbols	*IDR, +*
19	VB	Verbs that can display transitive verbs, intransitive verbs, active verbs, passive verbs, and copulas	*belajar, mengendarai, dimakan*
20	WH	A question word	*apa, bagaimana, siapa*
21	Z	Punctuation	*., ,, *, /, ?*

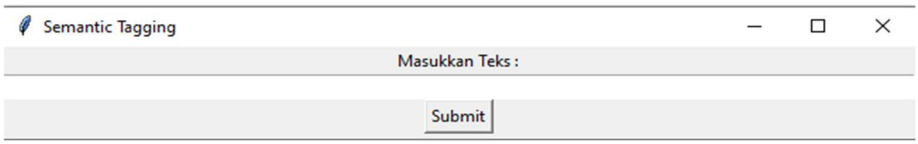

Fig. 3 Text Data Input Interface

18 anggota dewan positif corona , akankah gedung dewan perwakilan rakyat republik indonesia `` lockdown '' ? sebanyak 18 orang anggota dewan perwakilan rakyat republik indonesia dinyatakan positif virus corona . informasi tersebut mencuat usai dewan perwakilan rakyat menggelar rapat paripurna pengesahan omnibus law undang-undang cipta kerja . pada rapat yang diselenggarakan pada senin (5/10/2020) itu dihadiri 318 dari 575 anggota dewan perwakilan rakyat , baik secara fisik maupun virtual . sekretaris jenderal dewan perwakilan rakyat republik indonesia , indra iskandar , membenarkan kabar adanya anggota dewan perwakilan rakyat yang positif covid-19 tersebut . ia menyampaikan anggota dewan perwakilan rakyat yang terinfeksi covid-19 tengah melakukan karantina mandiri . `` ada 18 anggota dewan

Fig. 4 Example result of Semantic Tagging of Web Page Using DBpedia

18 anggota dewan positif corona , akankah gedung dewan perwakilan rakyat republik indonesia `` lockdown '' ? sebanyak 18 orang anggota dewan perwakilan rakyat republik indonesia dinyatakan positif virus corona . informasi tersebut mencuat usai dewan perwakilan rakyat menggelar rapat paripurna pengesahan omnibus law undang-undang cipta kerja . pada rapat yang diselenggarakan pada senin (5/10/2020) itu dihadiri 318 dari 575 anggota dewan perwakilan rakyat , baik secara fisik maupun virtual . sekretaris jenderal dewan perwakilan rakyat republik indonesia , indra iskandar , membenarkan kabar adanya anggota dewan perwakilan rakyat yang positif covid-19 tersebut . ia menyampaikan anggota dewan perwakilan rakyat yang terinfeksi covid-19 tengah melakukan karantina mandiri . `` ada 18 anggota dewan perwakilan rakyat yang terinfeksi , tapi juga ada dari

Fig. 5 Example result of Semantic Tagging of Web Page Using *schema.org*

```
<body>
    <div class="content">
        <a style="font-size:15px; font-family:Arial, Helvetica, sans-serif;"
property=" " resource="">18</a style="font-size:15px; font-family:Arial,
Helvetica, sans-serif;" property=" " resource="">
        <a style="font-size:15px; font-family:Arial, Helvetica, sans-serif;"
property=" " title="anggota dewan on DBpedia"
resource="http://dbpedia.org/resource/Board"
href="http://dbpedia.org/resource/Board">anggota dewan</a style="font-
size:15px; font-family:Arial, Helvetica, sans-serif;" property=" "
title="anggota dewan on DBpedia"
resource="http://dbpedia.org/resource/Board"
href="http://dbpedia.org/resource/Board">
```

Fig. 6 Example of Snippet HTML

Figure 3 shows the interface to submit documents. The example results of semantic tagging are shown in Figure 4 (with DBpedia) and Figure 5(with *schema.org*). The site fragment obtained in Figure 6 shows that semantic markup from DBpedia (bold) has been integrated, as shown below.
<div...resource=http://dbpedia.org/resource/Board href=http://dbpedia.org/resource/Board> anggota dewan <a style="font-size:15px; font-family:Arial, Helvetica, sans-serif;" property=" "...>

III. Results and Discussion

A. Dataset

The data is collected from the Indonesian language corpus, DBpedia, *schema.org*, and will be semantically labeled (Data Training). The Indonesian language database uses the fourth edition of the Large Indonesian Dictionary (KBBI), which includes 616,125 words with a file size of 8.71 MB in PDF format and 3.90 MB in .txt format. KBBI data includes entries, tags, pronunciation pointers, interpretation of meanings, usage examples, derivations, and word combinations. In DBpedia data, this job retrieves data that has resource types and properties. Resource types include 7,321,000 data with details of 4,233,000 resource data, 735,000 location data, 1,450,000 person data, 411,000 residence data, 251,000 species data, and 241,000 data about the organization. Meanwhile, the property type consists of 9993 data with a size of 16.1 GB in TTL file format. In the schema.org data, this work takes up 1,290 data layers with a size of 1.25 MB using the Resource Description Framework (RDF) data model.

An example of the dataset used is ***gotong royong (V)*** *bekerja bersama sama tolong menolong bantu membantu masyarakat berhasil membangun sebuah mesjid yg megah secara menghidupkan dan memperkembangkan dasar di desa desa* ***bergotong royong (V)*** *bersama sama mengerjakan atau membuat sesuatu* ***kegotong royongan (N)*** *hal bergotong royong*. V (Verb) is a verb that describes an action or activity. N (Noun) is the name of a noun. Examples of DBpedia classes are Machine (resource), Parent (resource), and Call (Property). The data will be semantically labelled, including 55 documents from various sources with two text types: 28 electronic news and 27 electronic articles. In addition, each piece of data is categorized into ten main topics: Education, health, economy, entertainment, automotive, lifestyle, tourism, environment, knowledge, and technology. Table 5 shows the example terms in DBpedia and *schema.org*.

TABLE V
EXAMPLE OF CLASS DATA FROM DBPEDIA AND *SCHEMA.ORG*

No	DBpedia		*schema.org*	
	Word	**Type**	**Word**	**Type**
1	Machine	resource	Thing	resource
2	Parent	resource	ChooseAction	property
3	Calls	property	IgnoreAction	property

A. The Results

This work used the fourth edition of the Big Indonesian Dictionary (KBBI) and was processed using an n-gram language model with lengths of two (bigram) and three (trigram). If there is a word and the prediction of the word after it that is entered is the same as that previously found in the training data and meets the threshold number, then the group of words is identified as a phrase with one meaning. Some of the threshold numbers tested get accuracy values as shown in Table 6 below:

TABLE VI
THRESHOLD ACCURACY VALUE

Trigram				Bigram			
T=0.9	**T=0.8**	**T=0.7**	**T=0.6**	**T=0.003**	**T=0.005**	**T=0.005**	**T=0.001**
0.38	0.67	0.63	0.63	0.61	0.63	0.61	0.56

Table 6 calculates the accuracy of the tested thresholds relative to the amount of data successfully obtained under each threshold. The determination of the tested threshold values is the result of experiments performed by the researchers by entering several possible phrases, both bigrams and trigrams. The phrase will generate a probability value, and the researcher gets an initial value of 0.7 for trigram and 0.003 for bigram. In addition, the researcher experimented with several values above and below the values found previously to reinforce the threshold value. Values are about 0.1 for Bigram because the number of Bigram phrases is not too much, so a high threshold value is used. At the same time, the values tested in the bigram are varied because slight differences give quite different exact results, and there are many phrases in the bigram.

This work uses a threshold value of 0.8 for trigrams and a threshold of 0.003 for bigrams because the mean precision values in these two trials are the highest. If a consecutive tuple is not defined in a bigram or trigram, it is automatically identified as a unique word or unigram data. Therefore, unigram recognition does not require a threshold value. From the threshold numbers used, this work successfully determined unigram, bigram, and trigram for 55 text data with the following percentage shown in Table 7. Table 8 for an example of the accuracy obtained for each document.

TABLE VII
PERCENTAGE OF OBTAINED UNIGRAM, BIGRAM, AND TRIGRAM

Unigram		Bigram		Trigram	
Frequency	Percentage	Frequency	Percentage	Frequency	Percentage
12742	93.2%	912	6.7%	15	0.1%

Based on the work results, some groups of words were identified as phrases because of their relatively high frequency of occurrence. However, semantically, a phrase either does not make sense or needs to be more appropriate to be a phrase. The rate of successful tagged data is 74% with DBpedia and 5.6% with *schema.org*. Data values can be from two significantly different knowledge bases. A lower score for schema.org happens because schema.org only has data related to the term.

Meanwhile, DBpedia covers a vast and varied range of data. Parts of the two KBs used can be Properties in DBpedia, like Actions in schema.org, and Resources in DBpedia, like Things in *schema.org*. Overall, the evaluation results use accuracy for the DBpedia knowledge base of 84.95% and 97.46% for *schema.org*.

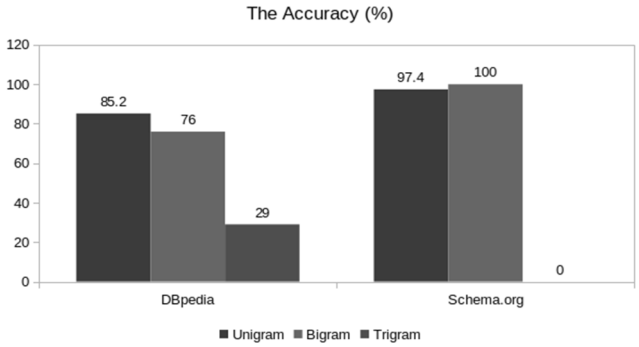

Fig. 7 Percentage of Unigram, Bigram, and Trigram Accuracy Values

The results of the exact values based on unigram, bigram, and trigram using two knowledge bases, DBpedia and *schema.org*, are shown in Figure 7. Figure 8 shows the results of the precision values of each type. Of the exact scores for each text category, the top score achieved in the lifestyle category was 90% for DBpedia, and in the entertainment and environment category, it was 100% for *schema.org*. Also, the lowest score for the auto category was 82.26% for DBpedia, and the auto category was 93.33% for *schema.org*.

The highest data distribution was 85%—89 % with 17 data on DBpedia and 90%—100% with 19 data on *schema.org*. The accuracy of *schema.org* results can reach 100%, as missing data can lead to ambiguity. Meanwhile, DBpedia's extensive data is more likely to cause ambiguity than *schema.org*. Most of the ambiguity in DBpedia is due to naming something that resembles the intended literal word.

The proposed algorithm successfully provides semantic markup of document data in Indonesian. The n-gram language modeling method used can also identify phrases well. However, the accuracy of the passphrase is highly dependent on the training data used. The more and better the training data is used, the better the system will identify phrases that match the semantics better. In addition, some words included in the list of stop words can make sense if they form phrases with other words. The data content used to provide POS tagging cannot be identified, which could lead to errors in the tagging process.

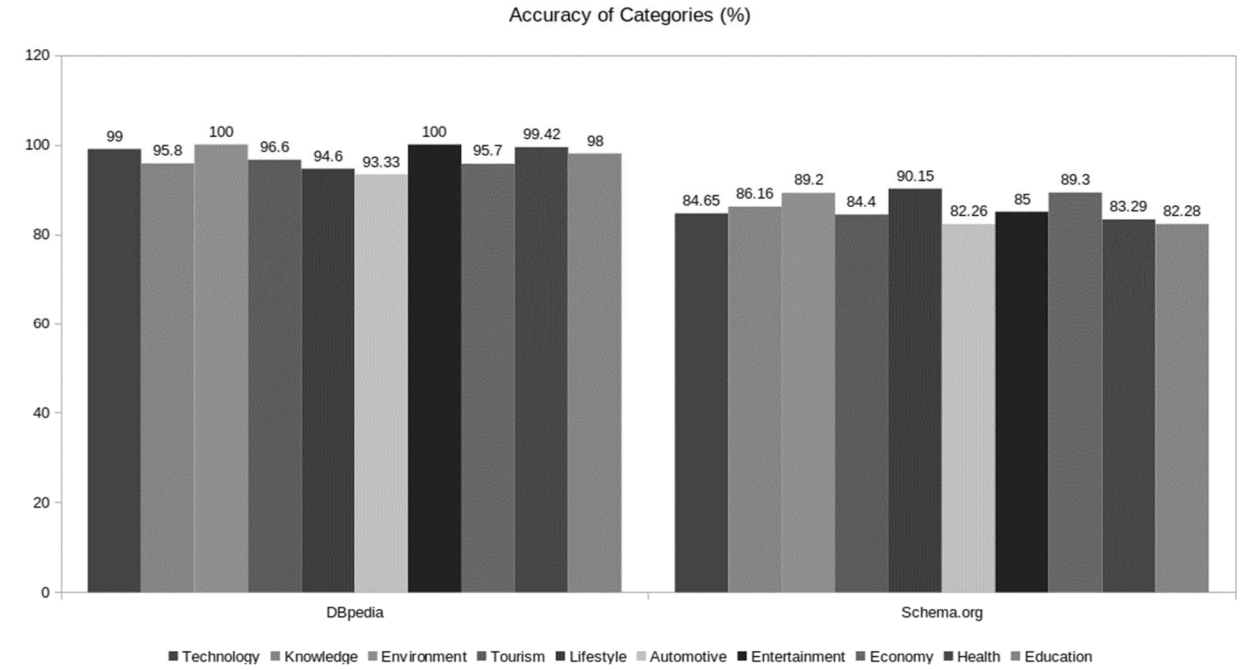

Fig. 8 Accuracy Value of Each Category

TABLE VIII
ASSESS THE ACCURACY OF EACH TEXT OF EXAMPLE DATA USING DBPEDIA AND *SCHEMA.ORG*

No	Title	DBpedia						*schema.org*					
		TT	TF	FT	FF	Total	Accuracy	TT	TF	FT	FF	Total	Accuracy
1	Pendidikan Indonesia di Tengah Pandemi Covid 19	263	19	56	42	380	80%	20	-	-	-	20	100%
2	Berpacu Uji Vaksin Covid 19	257	14	59	27	357	79.5%	45	-	-	-	45	100%
3	Pemerintah Daerah Tak Boleh Gegabah Buka Sekolah	216	23	40	17	296	78.7%	24	-	2	-	26	92.3%
4	Jalur Puncak Padat, Polisi Siapkan Rekayasa Lalu Lintas	101	20	18	11	150	74,7%	9	-	1	-	10	90%

Furthermore, there are limits to words that can be translated in one translation process. In removing the article, several characters are part of a word that is deleted due to the presence of these characters in the stop-words. Conversely, if the article characters are not included in the stop-words, there will be articles that cannot be removed, affecting the tagging results.

Accuracy results are affected by some factors, including the availability of the required data with the data contained in the knowledge base and the system's capabilities in the POS tagging process; image affects the result; new terminology and local cultures not yet available in the existing knowledge base. Used, and name recognition problems. There are some caveats regarding semantic markup, especially those made using the DBpedia knowledge base, such that DBpedia still needs more data for new terms to be used by the public. Widely, such as covid-19. Some verbs cannot be found in the

attribute but can be found in the resource. Then, based on the knowledge base of *schema.org*, the obstacle that most affected the outcome of this work was the limited data available because *schema.org* places more emphasis on the results of this work.

The semantic markup presentation in this study contains a link that will lead the user to each of the knowledge base data used. This shows the relevance, accuracy, or truth between the expected data and the resulting tagging. Furthermore, the format generated in this study did not follow the standards used in semantic markup, namely the Resource Description Framework in Attributes (RDFa) and the JavaScript Object Notation for Data Linked (JSON-LD). The representation of semantic markup must contain no links and be in plain text but must have semantic markup stored in the metadata. This work can then be continued to obtain the result according to the standards used in semantic markup.

IV. Conclusion

This work applied the N-gram language model to provide semantic markup on Indonesian unstructured text data automatically. Automated semantic markup generation starts with preprocessor text and implements the n-gram language model. The N-gram language model is used to identify phrases from text data in the Indonesian corpus. The text data is then converted into tokens that separate each word and phrase. Text data will be tagged by word type, translated into English, removed from articles, and tagged with word meanings. This work tagged text in HTML XML format using DBpedia and in microdata format using schema.org. The percentage of data that managed to be tagged in this job was 74% with DBpedia and 5.6% with schema.org. The DBpedia knowledge base has a high rate because the data in DBpedia is vast in scope, while schema.org puts more emphasis on terms. Overall, the assessment results used accuracy for the DBpedia knowledge base of 84.65% and 97.5% for schema.org. After documents are tagged, machines will easily access the data to be processed for the following computation. It will be helpful in any subsequent processes.

Acknowledgment

We thank the UNS Grant for funding this research.

References

[1] C. Anadiotis et al., "Graph integration of structured, semistructured and unstructured data for data journalism," *Information Systems*, vol. 104, p. 101846, Feb. 2022, doi: 10.1016/j.is.2021.101846.

[2] B. Sejdiu, F. Ismaili, and L. Ahmedi, "A Real-Time Semantic Annotation to the Sensor Stream Data for the Water Quality Monitoring," *SN Computer Science*, vol. 3, no. 3, Apr. 2022, doi: 10.1007/s42979-022-01145-6.

[3] A. Patel and S. Jain, "Present and future of semantic web technologies: a research statement," *International Journal of Computers and Applications*, vol. 43, no. 5, pp. 413–422, Jan. 2019, doi:10.1080/1206212x.2019.1570666.

[4] P. Hitzler, "A review of the semantic web field," Communications of the ACM, vol. 64, no. 2, pp. 76–83, Jan. 2021, doi: 10.1145/3397512.

[5] L. Tamine and L. Goeuriot, "Semantic Information Retrieval on Medical Texts," *ACM Computing Surveys*, vol. 54, no. 7, pp. 1–38, Sep. 2021, doi: 10.1145/3462476.

[6] T. Shaik et al., "A Review of the Trends and Challenges in Adopting Natural Language Processing Methods for Education Feedback Analysis," *IEEE Access*, vol. 10, pp. 56720–56739, 2022, doi: 10.1109/access.2022.3177752.

[7] M. Wang, J. Yuan, Q. Qian, Z. Wang, and H. Li, "Semantic Data Augmentation based Distance Metric Learning for Domain Generalization," *Proceedings of the 30th ACM International Conference on Multimedia*, pp. 3214–3223, Oct. 2022, doi:10.1145/3503161.3547866.

[8] X. Chen et al., "Imagine by Reasoning: A Reasoning-Based Implicit Semantic Data Augmentation for Long-Tailed Classification," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 1, pp. 356–364, Jun. 2022, doi: 10.1609/aaai.v36i1.19912.

[9] J. Zhang, Y. Zhang, and X. Xu, "ObjectAug: Object-level Data Augmentation for Semantic Image Segmentation," *2021 International Joint Conference on Neural Networks (IJCNN)*, Jul. 2021, doi:10.1109/ijcnn52387.2021.9534020.

[10] S. Albukhitan, A. Alnazer, and T. Helmy, "Framework of Semantic Annotation of Arabic Document using Deep Learning," *Procedia Computer Science*, vol. 170, pp. 989–994, 2020, doi:10.1016/j.procs.2020.03.096.

[11] Y. Pu, Y. Han, Y. Wang, J. Feng, C. Deng, and G. Huang, "Fine-Grained Recognition With Learnable Semantic Data Augmentation," *IEEE Transactions on Image Processing*, vol. 33, pp. 3130–3144, 2024, doi: 10.1109/tip.2024.3364500.

[12] A. A. Kardan, M. Fani Sani, and S. Modaberi, "Implicit learner assessment based on semantic relevance of tags," *Computers in Human Behavior*, vol. 55, pp. 743–749, Feb. 2016, doi:10.1016/j.chb.2015.10.027.

[13] A. Iliadis, A. Acker, W. Stevens, and S. B. Kavakli, "One schema to rule them all: How Schema.org models the world of search," *Journal of the Association for Information Science and Technology*, Feb. 2023, doi: 10.1002/asi.24744.

[14] J. Shafi, R. M. Adeel Nawab, and P. Rayson, "Semantic Tagging for the Urdu Language: Annotated Corpus and Multi-Target Classification Methods," *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 22, no. 6, pp. 1–32, Jun. 2023, doi:10.1145/3582496.

[15] B. Bostanipour and G. Theodorakopoulos, "Joint obfuscation of location and its semantic information for privacy protection," *Computers & Security*, vol. 107, p. 102310, Aug. 2021, doi:10.1016/j.cose.2021.102310.

[16] N. Wahab et al., "Semantic annotation for computational pathology: multidisciplinary experience and best practice recommendations," *The Journal of Pathology: Clinical Research*, vol. 8, no. 2, pp. 116–128, Jan. 2022, doi: 10.1002/cjp2.256.

[17] A. L. Santos, G. Prendi, H. Sousa, and R. Ribeiro, "Stepwise API usage assistance using n -gram language models," *Journal of Systems and Software*, vol. 131, pp. 461–474, Sep. 2017, doi:10.1016/j.jss.2016.06.063.

[18] K. D. Goyal, M. R. Abbas, V. Goyal, and Y. Saleem, "Forward-backward Transliteration of Punjabi Gurmukhi Script Using N-gram Language Model," *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 22, no. 2, pp. 1–24, Dec. 2022, doi:10.1145/3542924.

[19] D. A. Dahl, "Natural Language Semantics Markup Language (NLSML)," Speech Processing for IP Networks, pp. 159–173, Feb. 2007, doi: 10.1002/9780470060599.ch10.

[20] B. Drury, R. Fernandes, M.-F. Moura, and A. de Andrade Lopes, "A survey of semantic web technology for agriculture," Information Processing in Agriculture, vol. 6, no. 4, pp. 487–501, Dec. 2019, doi:10.1016/j.inpa.2019.02.001.

[21] R. Yu, U. Gadiraju, B. Fetahu, O. Lehmberg, D. Ritze, and S. Dietze, "KnowMore – knowledge base augmentation with structured web markup," Semantic Web, vol. 10, no. 1, pp. 159–180, Dec. 2018, doi:10.3233/sw-180304.

[22] R. Yu, U. Gadiraju, B. Fetahu, O. Lehmberg, D. Ritze, and S. Dietze, "KnowMore – knowledge base augmentation with structured web markup," Semantic Web, vol. 10, no. 1, pp. 159–180, Dec. 2018, doi:10.3233/sw-180304.

[23] S. Cardoso et al., "Use of a modular ontology and a semantic annotation tool to describe the care pathway of patients with amyotrophic lateral sclerosis in a coordination network," PLOS ONE, vol. 16, no. 1, p. e0244604, Jan. 2021, doi:10.1371/journal.pone.0244604.

[24] I. N. P. Trisna and A. Nurwidyantoro, "Single document keywords extraction in Bahasa Indonesia using phrase chunking," TELKOMNIKA (Telecommunication Computing Electronics and Control), vol. 18, no. 4, p. 1917, Aug. 2020, doi:10.12928/telkomnika.v18i4.14389.

[25] J. Tian, J. Yu, C. Weng, Y. Zou, and D. Yu, “Improving Mandarin End-to-End Speech Recognition With Word N-Gram Language Model,” IEEE Signal Processing Letters, vol. 29, pp. 812–816, 2022, doi: 10.1109/lsp.2022.3154241.

[26] S. Diao, R. Xu, H. Su, Y. Jiang, Y. Song, and T. Zhang, “Taming Pre-trained Language Models with N-gram Representations for Low-Resource Domain Adaptation,” Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pp. 3336–3349, 2021, doi:10.18653/v1/2021.acl-long.259.

[27] S. Avasthi, R. Chauhan, and D. P. Acharjya, “Processing Large Text Corpus Using N-Gram Language Modeling and Smoothing,” Proceedings of the Second International Conference on Information Management and Machine Intelligence, pp. 21–32, 2021, doi:10.1007/978-981-15-9689-6_3.

[28] A. Chiche and B. Yitagesu, “Part of speech tagging: a systematic review of deep learning and machine learning approaches,” Journal of Big Data, vol. 9, no. 1, Jan. 2022, doi: 10.1186/s40537-022-00561-y.