❏   961

Encapsulation of semantic description with syntactic components for the Arabic language

Adel Al-Janabi, Ehsan Ali Al-Zubaidi, Radhwan Hussein Abdulzhraa Al Sagheer
University of Kufa, Iraq

Article Info	ABSTRACT
	The work presents new theoretical equipment for the representation of natural languages (NL) in computers. Linguistics: morphology, semantics, and syntax are also presented as components of subtle computer science that form. A structure and an integrated data system. The presented useful theory of language is a new method to learn the language by separating the fields of semantics and syntax.

Corresponding Author:

Ehsan Ali Al-Zubaidi
Department of Envorimental Planning
University of Kufa, Iraq
Email: ihsana.kareem@uokufa.edu.iq

1. INTRODUCTION

In natural language, two autonomous phenomena are intertwined in NL: Discrete (grammar) and analog (semantics). Grammar (morphology and syntax) can be described in the language of mathematics [1]. Morphological the forms of words allow them to be distinguished and attributed they and their combinations have different values that they are recorded in the semantic dictionary [2]. In this case, this article shows you how to do this. The morphology model contains rules construction of word forms; word form of representation on the word in the original form and its morphological with parameters made up of decimal digits. We have the same morphological parameters forms a lexical group [3]. The model of NL syntax is based on the following assumptions are made: Mental sentences, that of the n members of the sentence you can always make up (n − 1) phrases (syntactically related pairs of word forms), such that the members of the sentence form a connected graph; in particular, combining simple sentences forms the syntactic graph Snt = (X, Y) from sets of X (word forms) and Y (word combinations) [4]. The degree of connectedness of two-word forms is determined by their morphological forms (parameters). That million of phrases, the Arabic language grammar, as well as they are divided into three hundred syntactic relations. Direct products of lexical groups, which makes it easier to represent the graph Snt in-memory computer [5]. The proposed network grammar model Snt opens up the following opportunities in the study of NL and its implementation on a computer:

a)   Root trees from the Snt model generate simple tree sentences. Bypass pre-location-tree assigns it a sentence sequence; correctness of the feedback a technique known as parsing offers depends on the extent to which the principles of fragmentation are observed. Between these sentences, the segments (tops of branches) included in the sentence tree fragments in the corresponding field sentence-sequence [6].

b) Simple sentences are used to construct more complex sentences according to the rules. From the rules for building a Snt model the possibility of building such a model follows grammar that any sentence of Arabic language (for example, sentences that meet-in literature in Arabic) [7] generates Snt syntax elements; hence, there is a formal definition of syntactic well-formed sentence and corresponding recognition algorithm for such proposals [8].

c) Morphology model (transformation rules words) is presented in the morphological dictionary; the program that implements these rules forms computer model of morphology [9]. Receptions syntax model (analysis (algorithms for analysis and synthesis of sentences) are demonstrated. Sentence values in Snt syntax presented in the semantic dictionary in the form classes to implement the semantic model semantics through object technologies programming [10].

2. RESEARCH METHOD

In this work, we implement on the idea according to an autonomous and constructive a description of the syntax that generates plans for you in each plan of expression the content plan the semantic element-is encapsulated [11]. The language becomes a reflection elements of two sets: syntax and semantics. It seems that the only separation is complete the development of syntax, and semantics opens the way to formalize NL [12]. Otherwise, you will have to agree with statements like No one can formulate compile all the rules of Arabic grammar. In textbooks on General Syntax and cautiously indicated on the tree structure of the sentence and the graph used only for demonstration network sentence structures. I had to decide and distinguish a class of elementary sentences that have structure of a rooted tree, and then notice that all other sentence forms (with homogeneous members and complex sentences) are collected from simple sentences [13]. Many construction rules are syntactically correct combinations can be represented as rules of a context-free grammar (for example measures, expressions formed from agreed and inconsistent definitions). This fact was used to represent syntax rules NL through substitutions and parsing trees [14]. The sentence how much context-sensitivity one hundred is required to provide reasonable structures tour descriptions? (How much context-sensitivity is required to provide reasonable structural descriptions?) from indicates that is far from thoughts of a separate study of syntax [15].

2.1. Morphology

Morphology is a structure given by a triple (A, L1, F1), where A is the alphabet; L1 - original lexicon which is a finite set source word [16] over the alphabet A; F1 is the final set of the first morphological characters, which are two-digit decimal numbers [17]. Initial signs of a split are intersecting subsets; elements of each subset form a linear array called category (attributes). Categories in morphology the Arabic language is less than ten, and they are named codes: 10 (gender), 20 (number), 30 (case) and (power), the original features in categories coded as follows: 10 = (11, 12, 13), 20 = (21, 22), 30 = (31, 32, 33, 34, 35, 36) and 40 = (41, 42, 43), or in a more familiar form [18]. The original vocabulary is also broken down into non-intersecting repentant subsets - the original parts of speech; each initial part of speech D1 has its own a set of categories Ψ and a set of features f_1; at-the sign is the string f = "$\alpha1\alpha2 \ldots. \alpha k$" from the original features belonging to different categories from the list [19]. One of the features of f1 is called the initial sign. Words from D1 have an initial sign; the original word s1 is represented a string like (s1: f1); form s of the word s1 with the com f will be represented as (s1:f_) for example, house: 21133 = home. Parts of speech are named by codes: 01 – name noun, 02 – adjective and 07 - verb. Categories for features will be Ω= {21131, 21132, 21133, 21134, 21135, 21136, 22131, 22132, 22133, 22134, 22135, 22136} [20]. For individual entrepreneurs from categories 10, 20, 30 and 40 compiled 29 signs: (411121131, 411121132, 411121133, 411121134, 411121135...). The set s1: Ω= {s1: f|f $\in\Omega$} is called the morphological group of the word s1, and its elements are called forms of the word s1, or only word. Group s1: Ω for s1 consists of 12 words, and for individual entrepreneurs - from 29-word forms; it is easy to see that the elements of the set s1: Ω are different. The union of the sets s1: Ω over all s1 from D1 denoted by D and called part of speech [16]. Morphology can now be represented even verification (A, L,$\Psi\_(1)$,Ω_1), where vocabulary L is a union all parts of speech; $\Psi\_(1)$ - many categories; $\Omega\_1$many signs. Numerical signs in the original form of the word indicate the forms of the word on which the syntax and sentences of the language is built. For converting NL word forms into words with numerical with signs and vice versa, appropriate morphological rules; they, as is known, can find in word-formation dictionaries. Consider the word s1: $\alpha1\alpha2 \bullet \bullet \alpha k$. By definition division, the original feature αk belongs to some category Fk [21]. The morphology of NL has such a property that the morphological features all the rows $\alpha1\alpha2\bullet\bullet\bullet\alpha k$ appear, where αk runs through the elements of the array Fk; they form a set $\alpha1\alpha2 \bullet \bullet \alpha k-1$ Fk is the paradigm of the word s1 over the category Fk [22]. Word forms are grouped by paradigms. In the electronic dictionary, instead of elements algorithms (morphological rules) that generate their elements will be written. For example, in the line (21130 home, house) behind the paradigm 21130, the word house lists its elements. It is possible, instead of word forms, to write put their postfixes like this:

House 0115:213000   (1)

In the first part "house 0115" of the article (1) by word "House" indicates its grammatical attributes (01is code, 15 - masculine and inanimate); in the second part lists two morphological rules (separated by semicolons) by which the elements of the paradigm 21130 and 21230 are built. A two-digit number is assigned to the paradigm code the length of the variable part of word forms from this paradigm. To save memory, it makes sense to store a separate list of postfixes of paradigm elements, and in the entries of the dictionary indicate their ordinal numbers. Building vocabulary entries is a chore that can also be programmed. Let s be the usual form of the word s1 with the sign $\alpha1\alpha2 \dots \alpha k$, that is, s = s1: $\alpha1\alpha2 \dots \alpha k$. In linguistic phenomenon, morphology solves two problems: synthesis transition from s1: $\alpha1\alpha2 \dots \alpha k$ to s - and analysis - the transition from s to s1: $\alpha1\alpha2 \dots \alpha k$. To analyze a word form, it will be necessary to synthesize all elements of the set s1: Ω. Word form analysis is complicated by the fact that from the form of the word s it is practically impossible to accurately guess its original form s1, and the dictionary the article begins with the original form of the word s. Word form analysis is a laborious procedure; to increase its efficiency, you need to use various search techniques from the discrete analysis [23].

2.2. Syntax

Let D1, D2,. ... ... , Dq - codes of parts of speech (as variable and adverbs that form one lexical group) in the morphology $\mu = (A, L, \Psi\_1 [\![,\Omega]\!]\_1)$ for them the sets signs $\Omega\_1, \Omega\_1, \dots \Omega\_q$ respectively. If a $\in [\![\Omega]\!]\_1$, then a: Di denotes the set of words from Di possessing the attribute a, and is called the lexical group by feature a [24]. For example, 21135 (01) = 0121135 = {home, point, food,. ... .}.By Ω_i (Di), or [Di], we denote the set consisting of lexical groups Di by to all criteria a from Ω_i. For example, the set (01) consists of 12 lexical groups: 0121131, 0121132, 0121133, 0121134,. ... ... , 012236. By Λ_μ we denote the union of all [D1], [D2],. ... ... , [Dq]. Syntax building begins by selecting several pairs of lexical groups from the set $[\![\Lambda]\!]\_\mu$:

X1 * Y1; X2 * Y2; X3 * Y3; ... ... ; Xk * Yk   (2)

R denotes the union of works:

R = X1 ∗ Y1 ∪ X2 ∗ Y2 ∪ • • ∪ Xk ∗ Yk   (3)

The product of lexical groups Xi ∗ Yi (i = 1,. ... ... , k) is called a syntactic relation their elements are syntactically related word forms or phrases; in the phrase (x, y), the word x is the main member, y is the dependent member of the combination. For example, 0121131 ∗ 0121132={(home, fashion), (smell, smoke), (sky, blue),. ... .}.The digraph (L, R) specifies the Snt syntax. Digraph (L1, R1), where L1 consists of lexical groups (2), and R1 is from pairs (Xi, Yi), where i = 1,. ... ... , k, also specifies packed Snt syntax; let Snt1 = (L1, R1). By Snt (Λ_μ) we denote some the syntax (grammar) given in Λ_μ. Syntactic relations (3) are specified by pairs of morphological features f1 and g1;

f2 * g2; ... ... ; fk * gk   (4)

where the features fi * gi define the lexical group's Xi * Yi (i = 1,..., K), hence, immense of size, the syntax Snt is specified by a small set (about two hundred pairs) of features (4) from Ω_i. For example, f1 = 012131 and g1 = 012132, f2 = 012131 and g2 = 022131,f3 = 012132 and g3 =022132, then in the graph (A, L,$\Psi\_(1$),Ω_1) will be connected by (v, w) only those vertices v and w that belong to: 0121131 ∗ 0121132,0121131 ∗ 0221131, 0121132 ∗ 0221132. Each of the three sets contains hundreds of thousands of items—coherent (agreed) definitions. The root tree in the Snt graph is called a tree expression [25]. Among the lexical groups,

Two groups: a group of predicates GP and a group of subjects GS. If the root of a tree expression belongs to GP and is connected by an arc to a vertex from GS, then such a tree expression is called a tree clause. Traversing a tree-expression is called a sequence-expression (or merely an expression), traversing a tree-clause is called sentence-sequence (just a [18]). If in the grammar Snt (Λ_μ):

a)   The set L is the lexicon of the Arabic language, (3) - functions from the morphology of the Arabic language;

b)   Elements Xi ∗ Yi (i = 1, … , k) are word combinations (pairs of related word forms) admissible in the syntax of the Arabic language;

c)   Linked pairs of word forms in Arabic expressions form a rooted tree,

Then Snt (Λ_μ) should have a lot in common with the syntax of the Arabic language; therefore, Snt (Λ_μ) will be called the model of grammar of the Arabic language [7], [26]. It is not difficult to show the existence of

such a grammar Snt (Λ_μ) that sentences found in Arabic literature will be sentences in the grammar Snt (Λ_μ). (But in the grammar Snt (Λ_μ) there will be sentences of dubious meaning.). The Snt grammar will be used for proper construction and expression recognition in the Snt syntax. Snt is an open system, it new words and syntactically related word pairs may appear. In particular, the well-known problem of parsing a sentence is similarly formulated and is solved both in NL and in Snt grammar: syntactically correct in Snt grammar the sentence will be a syntactically correct sentence in the Arabic grammar. A sequence of word forms a syntactically correct structure in the Snt grammar sentence if the graph Snt contains a tree sentence generated by this set of word forms. Moreover, algorithms for finding a rooted tree generated by a given set of vertices are well known [27]. A strict formulation of the problem of parsing a sentence and the presence of an algorithm for it solutions in NL grammar is already worth a lot. An important concept in Snt grammar is SF, the syntactic form. Shown on Figure 1. The tree has bracket representation: through the vertices H(D,E(C(A),B)) and 0 (1, 2 (3 (5), 4)). The vertices of the tree H(D,E(C(A),B)) are lexical groups, so it generates sentence trees where the letter denotes a word form from the group designated with the same letter. Suggestions from Snt generated by a rooted tree from same syntactic form, therefore such trees are called SF. Moreover, SF is a rule that generates a class offers of a specific form. Suppose that i (i = 1,..., 5) of the tree with Figure 1 (SF1) is represented by the product of a pair features fi * gi. Syntactic links between tree vertices require that f3 = f4 = g2,f5 = g3; any six words with the indicated signs can be the vertices of the sentence, generated by SF1. By assigning different word forms to different vertices of SF1, you can build different sentences. Denoting by g (t) the word form t: g (form g of the word t), we can write out the form of sentences generated by SF1:

$$f1 (s1) (g1 (t1), g2 (t2) (g3 (t3) (g5 (t5)), g4 (t4))) \tag{5}$$

Expression (5) is conveniently represented in the form of two isomorphic trees:

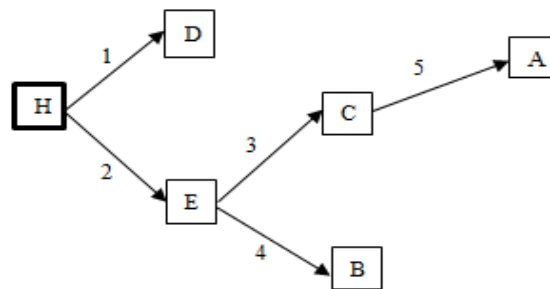$$f1 (g1, g2 (g3 (g5), g4)): s1 (t1, t2 (t3 (t5), t4)) \tag{6}$$

Figure 1. Root tree in Snt

it is easy to see that SF1 is represented by the expression by f1 (g1, g2 (g3 (g5), g4)). An example of SF is expression 011121131 (021121131, 011121132 (021121132)), which gives rise to expressions like the old hunter's white house. Form (6) will store expressions in the semantic dictionary. Studies of texts show that the number of different SF that generate simple sentences in Arabic, provided it is worth hundreds; three dozen SFs allow a native speaker of Arabic to express quite eloquently your thoughts and each author of texts has typical for him SF, which he uses to express your thoughts [10], [28].

2.3. Semantics

For all its content, formal grammar without semantics does not form a language. Semantics are based on syntax elements; syntax elements (words and their combinations) must express knowledge. Knowledge expressed by an element syntax is called it is meaning or semantics. Knowledge is a specific form of sensation consciousness of the active state of a specific area of human memory; witness the existing knowledge is a person who senses it; knowledge about a word (as a sequence of letters) let's call by his name the words are also stored in memory; another knowledge is attached to the knowledge of the word, called its value; man is able to reproduce word; the human perceived word activates its meaning; the word and its meaning are able to activate each other, this is the essence of NL [12]. The meaning and name of a word have the same relationship as information in a memory cell and the address of this cell, which is stored in another cell. Elements of syntax and semantics are manifestations in human physiology.

The relationship between two phenomena: sensation-word and sensation of knowledge meant about language as a reflection of two entities: elements of grammar (the plan of expression) and elements of semantics (the plan of the meaning). For digital technology capable of operating with elements of grammar, become a simulator language, it is required to find something in it (in technology), representing knowledge. For example, the functions of the plan of values could play neural networks (electronic circuits), if they consisted of them; but in modern computers, the organization of memory is such that the word on the screen the monitor and in the computer memory are entities of the same nature. Interlocutors using only elements syntax, exchange knowledge. Such communication is available to two computers if they are endowed with the same model of the world (the knowledge network) and identical transformation rules knowledge into sentences and vice versa. Knowledge in the model of the world can be stored in machine memory and the form of syntax elements. It remains to come up with a storage technology similar to that observed in the language ability of a person, and not in the articles of the explanatory dictionary. The following is one of the possible options building a semantic vocabulary that allows storing large amounts of knowledge in a convenient form for software processing [6], [29]. The entry V of the dictionary stores elementary knowledge about the concept of V in the form of a rooted tree T (V), in which semantic relations are displayed between V and other concepts. With a tree T (V) there is an SF to convert T (V) to an elementary syntactically correct expression. The syntactic form linked to a tree T (V), we denote by SF (T (V)). Semantic the dictionary consists of pairs hT, SF (T) i, which were are described in (6) and are suggestions of the Snt syntax. So the syntax elements are used to represent knowledge. Using the example of an article on the concept stork as shown in Figure 2, consider the structure of the article itself and evaluate what opportunities lie in such a dictionary for the formation of a language.

Figure 2. Semantic dictionary article describing the concept of stork

3. RESULTS AND DISCUSSIONS

The structure consists of two parts: a heading and a body. The name of the concept is followed by its full code - 04063, where 04 and 0406 are codes nested inside each other two ancestors (animal and bird) stork concept. Behind the slash indicates the name of the parent concept (bird) and part of speech code (01) of the concept of the stork. The body consists of five sections: shape, property, element, event, and method. The section shape lists the semantic forms of the word stork; in the section property the quality of this concept (colour, weight and shape); in the section element the constituent parts that form the described concept.

The event section lists the actions and states that the described concept can be subject to. The section method indicates the actions which the stork can perform over other concepts. In each section, knowledge is fixed, presented in the form of a semantic tree in parentheses; the root for all trees is the name of the described concept. Trees are prescribed SFs convert them to Snt syntax expressions. For example, the entry SF1: colour (white) can denote 011121131 (011121132 (021121132)): stork (colour (white)), that is, the expression white stork. Further, to construct a complex sentence from elementary can use rich experience syntactic and logical semantics.

4. CONCLUSION

Lexical and semantic analysis of natural language texts is intended to establish statistical patterns of occurrence names of concepts, service words, turns. The patterns obtained as a result of such analysis allow not only automatically recognize named entities, but also use call them to establish a system of relationships concepts in the formation of preliminary dictionaries of paradigmatic and associative links.

REFERENCES

[1]   M. Ibrahim and R. Ahmad, "Class diagram extraction from textual requirements using natural language processing (NLP) techniques," in *2010 Second International Conference on Computer Research and Development*, pp. 200-204, 2010, doi: 10.1109/ICCRD.2010.71.

[2]   E. Pons, L. M. Braun, M. M. Hunink, and J. A. Kors, "Natural language processing in radiology: a systematic review," *Radiology,* vol. 279, pp. 329-343, 2016, doi: 10.1148/radiol.16142770.

[3]   M. d. P. G. Mayo and N. Zeitler, "Lexical language-related episodes in pair and small group work," *International Journal of English Studies,* vol. 17, pp. 61-82, 2017, doi: 10.6018/ijes/2017/1/255011.

[4]   J. Hewitt and C. D. Manning, "A structural probe for finding syntax in word representations," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4129-4138, 2019, doi: 10.18653/v1/N19-1419.

[5]   R. M. Krishna and C. S. Reddy, "Extractive Text Summarization Using Lexical Association and Graph Based Text Analysis," in *Computational Intelligence in Data Mining-volume 1*, ed: Springer, 2016, pp. 261-272, doi: 10.1007/978-81-322-2734-2_27.

[6]   F. Nazir, W. H. Butt, M. W. Anwar, and M. A. K. Khattak, "The applications of natural language processing (NLP) for software requirement engineering-a systematic literature review," in *International conference on information science and applications*, pp. 485-493, 2017.

[7]   C. Khoury Aouad Saliby, C. Dos Santos, E. Kouba Hreich, and C. Messarra, "Assessing Lebanese bilingual children: The use of cross-linguistic lexical tasks in Lebanese Arabic," *Clinical linguistics & phonetics,* vol. 31, pp. 874-892, 2017, doi: 10.1080/02699206.2017.1308554.

[8]   J. Li, D. Xiong, Z. Tu, M. Zhu, M. Zhang, and G. Zhou, "Modeling source syntax for neural machine translation," *arXiv preprint arXiv:1705.01020,* 2017.

[9]   H. Borer, "Morphology and syntax," *The handbook of morphology,* pp. 149-190, 2017, doi: 10.1002/9781405166348.ch8.

[10] G. Boleda, "Distributional semantics and linguistic theory," *Annual Review of Linguistics,* 2020, doi: 10.1146/annurev-linguistics-011619-030303.

[11] E. Cambria and B. White, "Jumping NLP curves: A review of natural language processing research," *IEEE Computational intelligence magazine,* vol. 9, pp. 48-57, 2014, doi: 10.1109/MCI.2014.2307227.

[12] T. Young, D. Hazarika, S. Poria, and E. Cambria, "Recent trends in deep learning based natural language processing," *IEEE Computational intelligenCe magazine,* vol. 13, pp. 55-75, 2018, doi: 10.1109/MCI.2018.2840738.

[13] M. Inkster, M. Wellsby, E. Lloyd, and P. M. Pexman, "Development of embodied word meanings: Sensorimotor effects in children's lexical processing," *Frontiers in Psychology,* vol. 7, p. 317, 2016, doi: 10.3389/fpsyg.2016.00317.

[14] K. Chowdhary, "Natural language processing," in *Fundamentals of Artificial Intelligence*, ed: Springer, 2020, pp. 603-649, doi: 10.1002/aris.1440370103.

[15] H. Wu, Y. Chen, N. Wang, and Z. Zhang, "Sequence level semantics aggregation for video object detection," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 9217-9225, doi: 10.1109/ICCV.2019.00931.

[16] S. Fischev, M. Puzdyryova, S. Dmitrienko, D. Domenyuk, and A. Kondratyuk, "Morphological features of dentofacial area in peoples with dental arch issues combined with occlusion anomalies," *Archiv EuroMedica,* vol. 9, pp. 162-163, 2019, doi: 10.35630/2199-885X/2019/9/1/162.

[17] M. W. Klinker, R. A. Marklein, J. L. L. Surdo, C.-H. Wei, and S. R. Bauer, "Morphological features of IFN-γ–stimulated mesenchymal stromal cells predict overall immunosuppressive capacity," *Proceedings of the National Academy of Sciences,* vol. 114, pp. E2598-E2607, 2017, doi: 10.1073/pnas.1617933114.

[18] W. B. Starr, "A preference semantics for imperatives," *Semantics and Pragmatics,* vol. 13, p. 6, 2020, doi: 10.3765/sp.13.6.

[19] N. Mainz, Z. Shao, M. Brysbaert, and A. S. Meyer, "Vocabulary knowledge predicts lexical processing: Evidence from a group of participants with diverse educational backgrounds," *Frontiers in Psychology,* vol. 8, p. 1164, 2017, doi: 10.3389/fpsyg.2017.01164.

[20] Z. Zhang, *et al.*, "Semantics-aware bert for language understanding," *arXiv preprint arXiv:1909.02209,* 2019, doi: 10.1609/aaai.v34i05.6510.

[21] H. Dai, Y. Tian, B. Dai, S. Skiena, and L. Song, "Syntax-directed variational autoencoder for structured data," *arXiv preprint arXiv:1802.08786,* 2018.

[22] Y. Al Amrani, M. Lazaar, and K. E. El Kadiri, "A novel hybrid classification approach for sentiment analysis of text document," *Int. J. Electr. Comput. Eng. (IJECE),* vol. 8, pp. 4554-4567, 2018, doi: 10.11591/ijece.v8i6.pp4554-4567.

[23] G. Cassani, Y.-Y. Chuang, and R. H. Baayen, "On the semantics of nonwords and their lexical category," *Journal of Experimental Psychology: Learning, Memory, and Cognition,* vol. 46, p. 621, 2020, doi: 10.1037/xlm000074.

[24] M. Mesgar and M. Strube, "Lexical coherence graph modeling using word embeddings," in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2016, pp. 1414-1423, doi: 10.18653/v1/N16-1167.

[25] J. Liang, Y. Xiao, Y. Zhang, S.-w. Hwang, and H. Wang, "Graph-based wrong IsA relation detection in a large-scale lexical taxonomy," in *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, pp. 1178-1184, 2017.

[26] M. R. Rahman, M. T. Habib, M. S. Rahman, G. Z. Islam, and M. A. A. Khan, "An exploratory research on grammar checking of Bangla sentences using statistical language models," *International Journal of Electrical and Computer Engineering (IJECE),* vol. 10, p. 3244, 2020, doi: 10.11591/ijece.v10i3.pp3244-3252.

[27] I. El Bazi and N. Laachfoubi, "Arabic named entity recognition using deep learning approach," *International Journal of Electrical & Computer Engineering (IJECE) (2088-8708),* vol. 9, 2019, doi: 10.11591/ijece.v9i3.pp2025-2032.

[28] K. Arun and A. Srinagesh, "Multi-lingual Twitter sentiment analysis using machine learning," *International Journal of Electrical & Computer Engineering (IJECE) (2088-8708),* vol. 10, 2020, doi: 10.11591/ijece.v10i6.pp5992-6000.

[29] K. Mrhar and M. Abik, "Towards optimize-ESA for text semantic similarity: A case study of biomedical text," *International Journal of Electrical & Computer Engineering (IJECE) (2088-8708),* vol. 10, 2020, doi: 10.11591/ijece.v10i3.pp2934-2943.