The International Journal of Engineering and Science (IJES)
|| Volume || 8 || Issue || 11 || Series III || Pages || PP 42-48|| 2019 ||
ISSN (e): 2319 – 1813 ISSN (p): 23-19 – 1805

On the Automated Entity-Relationship and Schema Design by Natural Language Processing

Asst. Prof. Mohammad Kasra Habib
Balkh University Faculty of Computer Science

--**ABSTRACT**---
Diagrams play an important role in the software development process. Manual drawing of diagrams is a time-consuming task. Entity-Relationship (ER) diagram plays an extremely crucial function in the software development process and database design among other diagrams. Entity-Relationship data modeling is a high-level conceptualization that describes information as entities, attributes, and relationships. This type of modeling is to facilitate database design. There are many tools to draw the Entity-Relationship diagram manually. This paper describes the nature of natural languages and how to use Natural Language Processing (NPL) to design Entity-Relationship and schema under an automated process. Furthermore, it includes a terse overview of recent developments and discriminates among rule-based and probabilistic models.

Index Terms:*Entity-Relationship (ER) Model, Artificial Intelligence (AI), Entity-Relationship Diagram (ERD), Requirement analysis, Data modeling, System modeling, Information modeling, Natural Language Processing (NPL)*

Date of Submission: 25-11-2019 Date of acceptance: 07-12-2019

I. INTRODUCTION

Importance of diagrams and modeling are known in fields of engineering, particularly in software engineering. There are many diagrams, i.e., block diagrams, organizational chart, network diagram, pie chart, flow chart or ER-diagram.

Entity-Relationship (ER) models have played a central role in systems specification, analysis and development. Moreover, ER models are used to control and monitor system's databases. In ER modeling, a system's data is modeled as a set of entities, which is composed of a set of attributes, with their relationships [4]. However, obtaining Entity-Relationship models from a system's specifications may be a boring and time consuming process.

The idea behind this paper is to study recent researches which have been focused on automating the extraction of information from natural language text using Natural Language Processing (NPL). This process requires large amount of domain knowledge [1]. Generally, NPL employed to automatically convert information stored in natural language to a machine understandable format. The main goal of NPL is to extract knowledge from unstructured data that are highly ambiguous with complex grammars to be processed [2]. Natural language processing is a field of increasing importance with growing applications such as search, machine translation, and general human-computer interaction [3]. It is also a field in computer science and linguistics that is related to Artificial Intelligence (AI) and Computational Linguistics (CL). It is essential to have a review of Entity-Relationship Model (ERM) and literature. The details of having automated ER and schema through processing natural language is covered in upcoming sections.

II. OVERVIEW OF DATA MODELING

Models are created based on data. Therefore, the levels of logical views of data should be identified which the model is concerned [4]:

- Information concerning entities and relationship which exist in our minds.
- Information structure--organization of information in which entities and relationships are represented by data.
- Access-path-interdependent data structure---the data structures which are not involved with search scheme or indexing schemes.
- Access-path-interdependent data structure.

III. THE ENTITY-RELATIONSHIP DIAGRAM

Heuristics to Identify ER Elements (i.e., concept of entity, relationship, types and role). Take for instance there are two entities, both of them are of the “person” type (Figure 1). There is a relationship called “is-married-to” between these two persons. In this relationship, each of these two person entities has a role [5]; one of them plays the role of “husband”, and the other plays the role of“wife”.

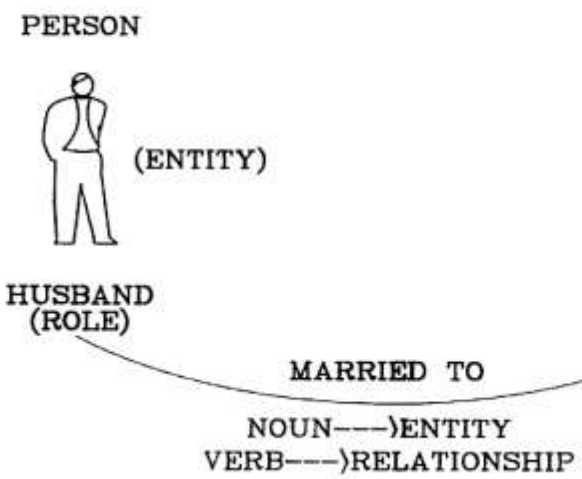

Fig. 1: Entity and relationship [5]

A key technique for ER modeling is comprehensive documenting the entity and relationship types with a graphical form called Entity-Relationship (ER) diagram [4]. Figure 2 shows a sample of entities with there relationships. The entity types such as EMP (employee) and PROJ (project) are placed at rectangular boxes, and the relationship types such as WORK-FOR are depicted as a diamond-shaped box [12]. The value sets such as EMP#, NAME, and PHONE are depicted as circles, while attributes are the “mappings” from entity and relationships types to the value sets [5]. The cardinalities information of relationship is also expressed [12], for example, the “1” or “N” on the lines between the entity types and relationship types indicated the upper limit of the entities of that entity type participating in that relationships.

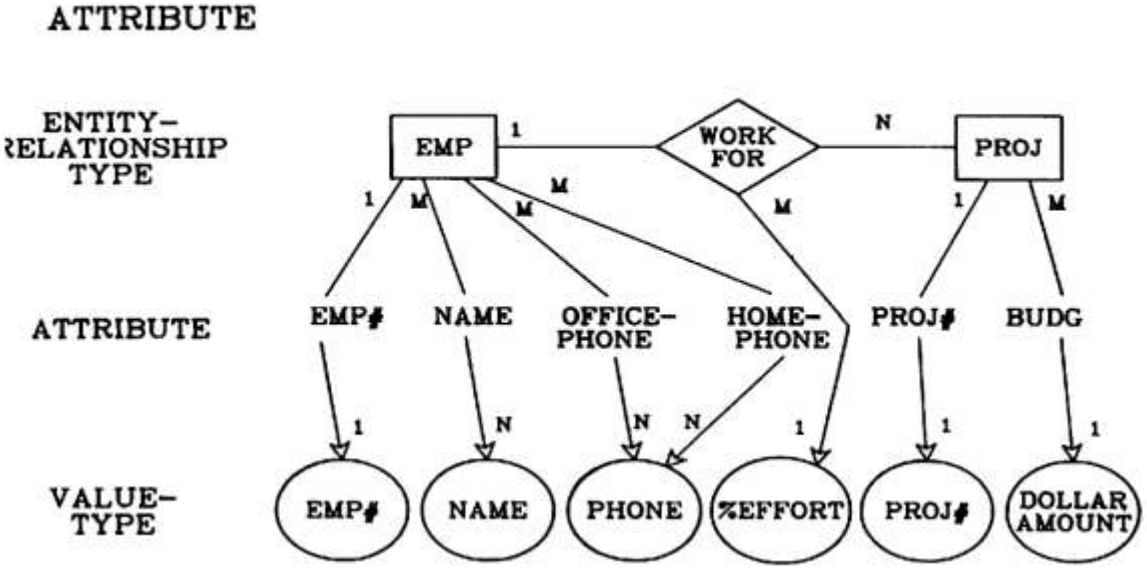

Fig. 2: An example of Entity-Relationship (ER) diagram [5]

3.1 Entity-Relationship Models are Based on Strong Mathematical Foundations

The ER model is based on [5]:

- Set Theory
- Mathematical Relations
- Modern Algebra
- Logic
- Lattice Theory

IV. LITERATURE REVIEW

The first phase in designing a database application is requirement analysis. It helps in understanding all the important data that must be stored in the database. This information is then conceptualized into high-level description of data [11]. This is done by designing the Entity-Relationship model.

An ER model can be thought of as a blueprint of data which will help us to understand the complexities of a functional system. ER models facilitate interaction among system analysts, designers, application programmers and end users. The concept of entity, relations, types and roles described in previous section. Therefore, main components of ER model are entity, attributes, relations, and carnality . To convert the users natural language it is necessary to use natural languages' grammar. ER components can be equated to parts of speech, as Peter Chen did. Comparison of ER diagram to grammar diagram [16] are as follows:

- Common noun (entity type)
- Proper noun (entity)
- Verb (relationship type)
- Adjective (attribute for entity)
- Adverb (attribute for relationship)

V. NATURAL LANGUAGE PROCESSING (NPL), AUTOMATED ENTITY-RELATIONSHIP AND SCHEMA DESIGN

Natural language casting to ER is discussed in foregoing section. Therefore, to do all these steps automatically we need to use Natural Language Processing NPL. Natural Language Processing (NPL) is one of the central goals of AI. NPL can be used to achieve automation for generating ER diagram [3]. Lots of researches have been accomplished in application of structural analysis for generation of ER diagram. In-addition by help of NPL, there are two fashion to map from natural language to conceptual design, rule- and probability- based mapping (visit [17] for more details on rule- and probability- based mapping), which both ways have their advantages and disadvantages.

5.1 Rule-Based Mapping

Rule based design tools maintain rules and heuristics in several knowledge bases [19]. A parsing algorithm which accesses information of a grammar and a lexicon is designed and integrated to meet the requirements of the tool [5, 19]. During the parsing phase, the sentence is parsed by retrieving necessary information from the grammar, represented by syntactic rules and the lexicon. The parsing results are processed further on by rules and heuristics which set up a relationship between linguistic and design knowledge.

5.2 Probability-Based Mapping

Rule based tools translates all verbs, nouns to entities and relationships. It may not be appropriate to translate all verbs into relationships, or entities to nouns as it does not hold true for all cases. To overcome the limitations of rule-based model researchers uses n-gram model which is a probabilistic language model [20].
N-grams are widely used language model; relies on the fact that the probability of one word in a document depends on its previous n-1 words [18].

VI. APPLYING NATURAL LANGUAGE PROCESSING TO ACHIEVE AUTOMATED ERD AND SCHEMA

The typical architecture for generating ER from natural language is information extraction. The process begins by sentence segmentation processing, which is a morphological analysis applied to specifications followed by tokenization process; results from this process are words only [6]. Part Of Speech (POS) process tags each word with its abbreviations [21]. Chunking and parsing apply multiple possible analyses on results [22]. Parsing is the process of using a grammar to assign a syntactic analysis to a string of words forming parsing tree. Finally, extracted information from parsing tree is used to generate ER diagram [6]. Each process is described in detail in the following subsections, which is being discussed in upcoming titles.

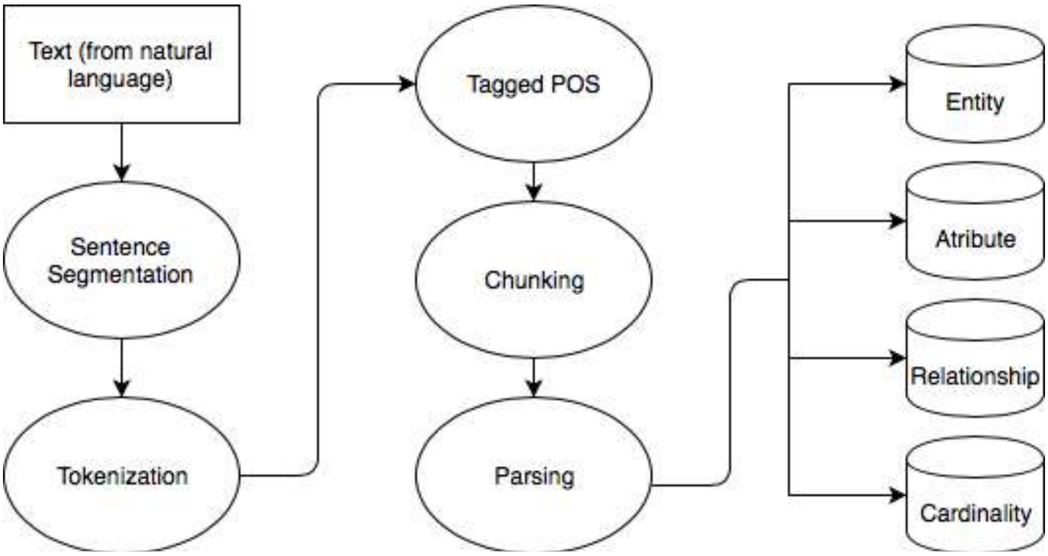

Fig. 3: Natural Language Processing engine for ERD

6.1 Sentence Segmentation

This step, morphological analysis is applied on the natural language text. User/Operator gives the requirements specification as input to NPL system. Then, the system performs analysis and split the text into sentences, since each sentence must end with period it is easy for system to understand [22]. Eliminate all non-word tokens like punctuation, removing plural suffixes in nouns, i.e., "s", "es" or "ies", and converting plural entity names into singular.

6.2 Tokenization

In tokenization process, words and numbers in each sentence are identified. It is necessary to specify the sentence's components. Basically, the proposed tokenization is set to break up the given sentence into units called tokens separated by spaces. For example, the sentences "It is not a punishment". The tagged sentences appear as < its>< not>< a><punishment>. Such implementation similar to string.split (' '),in programming languages. Tokenization process can identify each word in user input data [21]. However, compound words that use commas and periods add complexity [6]. For example, a tokenizer may have to recognize that the period in "Mr. X", the dot after Mr. does not terminate the sentence.

6.3 Tagged Part Of Speech (POS)

Part Of Speech (POS) tagging is the process of identifying a word in a text as corresponding to a particular part of speech, based on its definition and context [23]. Table 1 is a summary of symbols and abbreviations [6]. For example, tokenize the following sentences, "The little X saw Y with a crazy dog recently" is {the/ Article, little/Adjective, X/Noun, saw/Verb, Y/Noun, with/Preposition, a/Article, crazy/Adjective, dog/ Noun, recently/ Adjective}.

Table 1: List of symbols and abbreviations [23]

Symbol	Abbreviation
SS	Sentence
Adj	Adjective
NP	Noun phrase
V	Verb
PN	Proper noun
Prep	Preposition
Art	Article
N	Noun
Pro	Pronoun
VP	Verb phrase
Adv	Adverb
PP	Prepositional pronoun

6.4 Chunking

Chunking is the process of taking individual units of information (chunks) and grouping them into larger units [6]. Tokens of a sentence are group together into larger chunks, each chunk corresponding to a syntactic unit such as a noun phrase (NP) or a verb phrase (VP). To perform the chunking, a Part of Speech

(POS) tagged set of tokens are required with tokens itself. POS tagging tells whether words are nouns, verbs, adjectives and etc. At this step the sentences are converted to this form “We saw the yellow dog” is {We/NP, the yellow dog /NP}. Another example for the sentence “X bought Porsche” is {X/N, Porsche/NP bought Porsche/VP}. Also, chunk the sentence “X hit the ball” is {X/NP, the ball /NP, hit the ball /VP}.

6.5 Parsing

Parsing process determines the parse tree of a given sentence [6]. Since natural languages grammar is ambiguous and has multiple possible analyses. Each sentence may have many potential parse tree. The words are transformed into parse tree structure to understand how units of sentence's are related to each other (Figure 4).

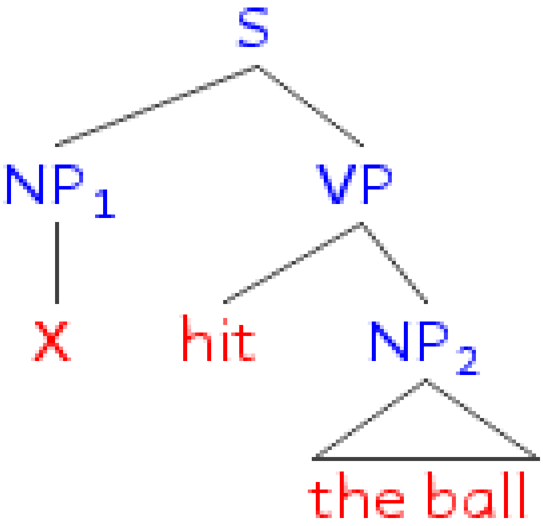

Fig. 4: Parser tree for “X hit the ball”

The proposed methodology based on a set of identification rules that combines different concepts from other works as follows:

Rule 1 identifying entities:

- A common noun may indicate an entity type [7, 8]
- A proper noun may indicate an entity [7, 8]
- A gerund may indicate an entity type [7]
- A specialization's relationship “A is a B” sentence's structure can relate two nouns [9]
- A noun such as “database”, “record”, “system”, “information”, “organization” and “detail” may not be considered as a candidate for an entity type because it shows the business environment [10]

Rule 2 Identifying attributes:

- Noun phrase with genitive case may indicate an attributes [8]
- If a noun is followed by another noun and the latter one belongs to set S where S= [number, no, code, date, type, volume, birth, id, address, name], this may indicate that both nouns are an attribute else it may be an entity [10]
- A noun such as “vehicle no”, “group no”, “person id” and “room type” refer to an attribute [11]
- The possessive case usually shows ownership it may indicate attribute type [8]
- A noun phrase such as “has/have” may indicate attribute types [11]

Rule 3 Identifying relationships:

- A transitive verb can indicate relationship type [7]
- A verb followed by a preposition such as “by”, “to”, “on” and “in” can indicate a relationship type [8]

If a verb is in the following list {include, involve, consists of, contain, comprise, divided to, embrace}, this indicate a relationship of aggregation or composition [21]

- An adverb can indicate an attribute for relationship [7]

- A verb followed by a preposition such as {on, in, by, to} could be a relationship. Take for instance, {Persons work on projects}. Other examples include {assigned to} and {managed by} [10]

Rule 4 Identifying primary key:

- Adverb (uniquely) indicates Primary Key (PK) of an entity [13]
- If the sentence is in the form of {“Subject” + “Possessive verb” + “Adjective” + “Object”}, then the object is a key attribute [13]

6.6 Generating Entity-Relationship

The ER generator is a rule-based system that identifies ER relationships, ER entities and ER attributes [6]. Once all words have been assigned to its ER element type, relevant information consisting of which words are entities, relationships, cardinalities and attributes are stored in text files. These text files are then used to generate ER diagram.

VII. DISCUSSION (PERFORMANCE EVALUATION, RESULTS AND WHERE IS IT SUITABLE?)

Since now, this paper have argued about ERD, NPL and how its possible to implements ER diagrams using NPL to achieve an automated way of developing. There are tens of tools available like GetER [17], ER-Converter [6] and CM builderand-LOLITA [6] which are either type of rule based or probability based mapping. In-addition to check there completeness and correctness of them, they are tested across different scenarios.

7.1 Performance Evaluation

Evaluation of tools have been done based on manual mapping, rule based mapping and probability based mapping [17]. The goal behind this work is to implement a prototype to demonstrate ER modeling by using of NPL.

Rule based mapping gives good result for those statements which are of a specific format [17]. However there are many occurrence that rule based mapping fails to give output but probability based models can suggest recommendations [17]. Multi-sentence inputs also gives fair results if they are grammatically correct and in subject and object format [17]. The process has been done by providing single sentences as input text from different domain such as library management system, banking, hospital management, hotel or hostel systems. For 500 sentences it has carved out nearly 1048 unique triplets [17].

7.2 Results

Different scenarios are being tested with rule based module and probabilistic module. 500 sentences are used for training purpose and 100 sentences are used for testing purpose [17]. Results are checked with manual mapping, rule based mapping and probabilistic model mapping. From driving scores it can be concluded that probabilistic models are more promising (Table 2).

Table 2: Probability- and rule- based precision and recall scores

Category	Precision	Recall
Rule-based	86.9	86.3
Probability-based	91.3	89.0

7.3 Real World Applications of Automated Entity-Relationship Diagrams

Previous subsection have discussed about accuracy of automated ER design by NPL. The only question remind is where its suitable to use? It would be great to built systems by use of other advanced systems and make daily routines easy and fast by automation. According to results from Table 2, these systems are still under development due to complexity and ambiguity in natural language. There are available systems on internet to use for academic approaches and researches for those who are interested at this field.

VIII. CONCLUSION

To sum up, an entity-relationship model (ERM) is a theoretical and conceptual way of high level data modeling and their relationships in software development process which is widely used for database modeling. Recently many researchers have attempt to cast the old fashion (manual/classic) design procedures with an automated style with help of Natural Language Processing (i.e., tokenization, tagging POS, chunking and parsing based on syntax heuristics rules), to gain knowledge from requirements specification.

This approach of creating automated ER and schema designhas benefits such as, no need of having high knowledge to create ER and schema, easy and fast to build. However there are limitations like linguistic variation (incomplete knowledge) and understanding grammar (tagging of part of speech). Moreover, not to forget that NPL is still under development and call for further research to improve NLP engines using neural networks and advanced algorithms such as a-priori or Support vector Machine (SVM) and they are available only for academic purposes.

REFERENCE

[1]. S. Geetha, and G. A. Mala, "Automatic Relational Schema Extraction from Natural Language Requirements Specification Text", Middle-East Journal of Scientific Research, vol. 21, no. 3, (2014), pp. 525-532.

[2]. F. Hogenboom, F. Frasinca and U. Kaymak, "An Overview of Approaches to Extract Information from Natural Language Corpora," Information Foraging Lab, (2010), p. 69.

[3]. C. Andrews, "A Natural Language Interface Using First-Order Logic" A Major Qualifying Project Report: Submitted to the Faculty of(Doctoral dissertation, WORCESTER POLYTECHNIC INSTITUTE). (2005).

[4]. Peter Chen "The Entity-Relationship Model---Toward a Unified View of Data", Massachusetts Institute of Technology (MIT), pp. 10-18.

[5]. Peter P. Chen,"Entity-Relationship Modeling: Historical Events, Future Trends, and Lessons Learned", Computer Science Department Louisiana State University, pp. 3-4.

[6]. Ronak Dedhia, Atish Jain, Prof. KhushaliDeulkar. "Techniques to automatically generate Entity Relationship Diagram", IJIACS ISSN 2347 – 8616 Volume 4, Issue 10 October 2015.

[7]. P. Chen, “English Sentence Structure and Entity Relationship Diagrams”, International Journal of Information Science, vol. 29, (1983), pp. 127-149.

[8]. A. M. Tjoa and L. Berger, “Transformation of Requirement Specification Expressed in Natural Language into an EER Model,” Proceedings of the 12th International Conference on Entity Relationship Approach, Springer Verlag, New York, (1993), pp.127-149.

[9]. S. H. Sebastian, “Link.: English Sentence Structures and EER Modeling", In Proceedings of APCCM'2007. p. 27-35.

[10]. H. M. Harmain and R. Gaizauskas, “CM-Builder: An Automated NL-based Case Tool”, 15th IEEE International Conference on Automated Software Engineering (ASE’00), (2000), pp. 45-54.

[11]. V. G. Storey, “View Creation: An Expert System for Database Design, ICIT Press, (1988).

[12]. S. Geetha, G. S. A. Mala, “Automatic database construction from natural Language requirements specification text", ARPN Journal of Engineering and Applied Sciences, ISSN 1819-6608, vol. 9, no. 8, (2014), p. 1260-1266.

[13]. L. A. Al-Safadi, "Natural Language Processing for Conceptual Modeling", JDCTA, vol. 3, no. 3, (2009), pp. 47-59.

[14]. Btoush,EmanS.,andMustafaM.Hammad."Generating ER Diagrams from Requirement Specifications Based On Natural Language Processing." International Journal of Database Theory \& Application 8.2 (2015).

[15]. P. R. Kothari, "Processing Natural Language Requirement to Extract Basic Elements of a Class", ISSN.- 2249-0868 Foundation of Computer Science PCS, New York, USA, vol. 3, no. 7, (2012).

[16]. Elmasri, Shamkant B. Navathe, "Fundamentals of Database Systems", Pearson 2016.

[17]. Manasi. Patwardhan, Mugdha. Shah*, Preeti. Bailke, "Extraction of Conceptual Schema from Natural Language using Machine Learning", Vishwakarma Institute of technology, Pune-43, India.

[18]. Zhengxian Gong, Min Zhang Chewlim, Tan Guodong Zhou - “N- gram-based Tense Models for Statistical Machine Translation”, Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Learning, pages 276–285, Jeju Island, Korea, 12–14 July 2012

[19]. Sebastiani, Fabrizio, "Machine learning in automated text categorization", ACM computing surveys (CSUR), pages 1-47, Volume 34, 2002

[20]. Lewis, David D, "Feature selection and feature extraction for text categorization", Proceedings of the workshop on Speech and Natural Language, pages 212--217, 1992

[21]. Zheng, Xiaoqing and Chen, Hanyang and Xu, Tianyu, "Deep learning for Chinese word segmentation and POS tagging", Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, pages 647--657, 2013

[22]. Foster, Jennifer and Cetinoglu, Ozlem and Wagner, Joachim and Le Roux, Joseph and Hogan, Stephen and Nivre, Joakim and Hogan, Deirdre and Van Genabith, Josef, "hardtoparse: POS Tagging and Parsing the Twitterverse", Workshops at the Twenty-Fifth AAAI Conference on Artificial Intelligence, 2011

[23]. Habash, Nizar et. al., "MADA+ TOKAN: A toolkit for Arabic tokenization, diacritization, morphological disambiguation, POS tagging, stemming and lemmatization", Proceedings of the 2nd international conference on Arabic language resources and tools (MEDAR), Cairo, Egypt

Asst. Prof. Mohammad Kasra Habib "On the Automated Entity-Relationship and Schema Design by Natural Language Processing" The International Journal of Engineering and Science (IJES), 8.11 (2019): 42-48