Performance Evaluation of Automated Construction of Domain-Based Ontology Using CNN and Machine Learning

[1*]Giridhar Urkude, [2]Moon Banerjee, [3]Bala Subramanyam P N V, [4]B Lakshmana Swamy, [5]Mohan Awasthy

[1]Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Hyderabad, Telangana-500075, India.

[2, 3, 4]Department of Mechanical Engineering, Koneru Lakshmaiah Education Foundation, Hyderabad, Telangana-500075, India.

[5]Bharati Vidyapeeth Deemed to be University, Department of Engineering and Technology, Navi Mumbai

ABSTRACT

Ontology is essential to the Semantic web and many new AI applications. With the assistance of ontology, the client and the system may interact with the popular domain understanding in a machine-to-machine environment. While ontology has been proposed as an essential tool to reflect real-world information in database design construction, most ontological innovations are not immediately implemented. The fundamental challenges to developing and updating these Domain-specific Ontologies, such as the need for manual involvement by field experts and the limitations imposed by current technology, have made it less feasible for ontology to create and upgrade automatically. Automatic ontology generation plays therefore an essential role in semantical web applications and emerging AI applications. However, domain experts need manual involvement to help build and update the domain-specific ontology, and the limitations on existing technology adoptions make it less feasible to create and update ontologies automatically. The key contribution of this research work is to generate automated ontologies from discrete sources of knowledge based on machine learning algorithms. Study findings are in four distinct stages, data extraction from many sources, automated acquiring of information using machine learning algorithms and selection of automatic attributes, and the modelling and validation of the relationship between entities and ontology. A model is developed for acquiring and exploring information in field ontology using the Jaccard Relationship and the Neural Network. The results show that the machine-built models can construct domain-specific ontologies automatically and efficiently.

Keywords: machine learning, domain-specific ontology, attribute selection, Knowledge acquisition, automation.

1. INTRODUCTION

Ontologies are an important and comprehensive part of the Semantic Web; they are used as resources in web to represent information and to allow machines to comprehend internet data concepts. The ontological life cycle through the Semantic Web involves manual creation, refining, mixing, mapping, annotation techniques, etc., which involve the classification of core population principles, annotations, manipulation, or management of ontology. Ontology capacities semantic annotation applications to allow structural, syntactic, and semantic document definition, which provides entirely new ways of intelligent search rather than keyword matching, preference for query response over knowledge recovery, Ontology mapping practice for document exchange between applications, and document view definitions, at the same time it should be scalable in order to extend the idea by other users Ontology learning (OL) involves the importation, processing, extraction, cutting, and evaluation of ontology. Ontology is of the utmost importance when arranging information for use by integrated machine-by-machine (M2M) systems for explicit inferences, which is sufficiently integrated into the use of artificial intelligence (AI), natural language processing (NLP). Ontologies find application in terminology acknowledgment, user response understanding, and word generations for particular user interaction concepts and channels for discussion, such as chatbots, etc.

Notable examples demonstrate the impact of neural approaches to learning in the acquisition of information and representation in the broad field of Semantic Web technology. Litratures include ontology learning [1, 2, and 3], the learning of organized natural language query [4], the alignment of ontology [5, 6, 7], annotation of ontology [8, 9], combined relationship and multi-modal representations of knowledge [10], and prediction of relationships. Ontologies, by comparison, have been used repeatedly for machine-learning tasks as context information. For example, there are countless hybrid approaches to language learning by integrating corpus-based evidence with semantic sources [11, 12, 13, 14, and 15]. This interplay of formal information and corpus-based methods has led to knowledge-graphics embeddings, which have proved useful in tasks like the discovery of hypernymes [16], discovery and classification of collocation [17], meaning disambiguation [18, 19], and related relational and multi-modal representations of knowledge [20], and many more.

Domain-specific ontological study accords to the appropriate interpretation of the knowledge structure. Ontology is the basis for knowledge in the appropriate framework for any given domain. The use of ontology or conception determines the creation of the knowledge level and the vocabulary for this knowledge. In order to make this knowledge pervasive, the knowledge base (ontology) is created once and encouraged to either reuse this knowledge or, if concepts are not available, add to it and publish it as linked open data.

The application of Artificial Intelligence, Semantic Web, Biomedical Informatics, Systems Engineering, Enterprise Bookmarking, Information Engineering, and Library Science can be properly used for particular domain ontologies. These applications have contributed to the relevant understanding and perception of information and its role in the expression of the domain. For the following purposes, special domain ontologies are generated, which can be used advantageously for the adequate representation of knowledge:

- Ontology provides a common and sharable domain vocabulary.
- Ontology metadata enables easy fusion and expansion of ontologies i.e. scalability.
- Content is clearly defined by ontology.
- Knowledge of the domain can be separated from operational knowledge.
- Ontology enables the reuse of its contents.
- Ontology offers ordering and structuring of its contents.
- The addition of cognition capabilities and rules helps to infer new knowledge.

In the Semantic Web and numerous emerging applications, Ontology plays an important part. With the help of ontology, the client and the framework can interact with each other in a machine-to-machine environment with a common understanding of the domain. Although ontology has been proposed as a vital means of representing real-world knowledge in the construction of database designs, most ontological developments are not carried out automatically. However, the underlying challenges in creating and updating these Domain-Specific Ontologies, such as the need for manual intervention of Domain Experts and the restrictions imposed by current technology adoptions, have made the tasks of automatic creation and upgrading of Ontologies less feasible. The automatic generation of Ontology, therefore, plays a significant role in the semantic web and emerging AI applications.

1.2 Why build ontologies for a particular domain?

For five purposes referred to below, domain-specific ontologies are created,

1. Field expertise review,
2. Distinguishing domain experience and organizational expertise,
3. Domain assumptions are made clearly,
4. Enable domain information to be reused,
5. Sharing a summary for app users and agents of information systems,

1.3 Research objectives

- K-Bayes Ontology Learning Algorithm is designed which automatically extracts concepts, characteristics, values, and relations across fields (K-Bayes).
- The implementation of unregulated methods of representation training on ontologies that establish "“embedding”"" for entities in ontologies, and we demonstrate that these

embedding methods can be used as semantical measures of similarity while helping to overcome such limitations.

- Designing an algorithm for automatic entity and relationship validation.

2. RELATED WORK

A study of different ontological methods has been published by Hazman et al. [21]. The analysis of ontology was divided into two categories, unstructured and semi-structured data. The study found that the processing of natural language techniques was useful in learning ontology from unstructured data. In the case of ontology from semi-structured data, data mining and web content mining techniques are more important. These concepts explored ontology using domain keywords during their survey but did not explore the ontology of building from scratch. The study showed that ontology evaluation is both essential and significant. The five levels of ontology assessments were defined, namely the lexical, hierarchic, contextual, syntactic, and structural levels (vocabulary). It was concluded that human-based assessment is possible on all five levels listed above.

To extract domain documentation from the web and use them as a corpus in extracting terms and concepts for ontological construction, Sanchez and Moreno et al. [22] used seed words. Furthermore, Fraga and Vegetti [23] placed seed words in a text file to manually facilitate extraction. Current research relied primarily on NLP ontology to provide the system with context information. Normally, only a human expert solves the key problem of automatically learning complex domain ontologies. The first to apply neural language models to PubMed corpora were Pyysalo et al. [24] and Minarro-Gimenez et al. [25]. The use of Skip-gram 22-m PubMed and more than 672 K PubMed Central Open Access is the total text papers from Pyysalo et al. The principal goal of Pyysalo et al. work was to render word representations accessible (1 to 5 grams) from reusable literature. Minarro-Gimenez et al. have used PubMed and other medical and non-medical datasets.

The mixed paradigm used by clinicians to promote automated decision-making, Holmes (Hybrid Ontological and Learning Medical System) developed by Khan et al. [26]. Holmes integrates knowledge-based technology with ML that generates a decision support system resistant to noise (DSS) when complete information is missing. The lack of data management is an important issue for current medical DSS as patients provide inaccurate details or do not disclose current health conditions. It is therefore essential to have a forum that addresses these concerns. A specific construction was given for the validation of the suggested hybridized design in which the central reasoning used knowledge-based decision making, and the ML algorithms predicted the use of custom classifying. Holmes has incorporated a design feature that removes the need to plan complete datasets before processing. Holmes is also able to use many different data sets, which enables the sharing of information in real-time that is particularly useful in medical emergencies. The paradigm is however limited by the lack of exploration of models for machine learning for the decision-making process. It has also been developed for use in a particular situation and thus needs to be modified (or extended) to other applications.

A further recent research, Moran et al. [27] found that the processing of remote sensing information for the production of analytical and comparable evidence involves automated mechanisms. Study developed a technique to combine ontological knowledge management with ML, using methods for classification and regression trees to identify various spatial data sources. The principal advantages included promoting skillful information management and logical skills in ontology, such as consistency monitoring. The ontology created can also be used or moved to the Linked Data Cloud in other investigations. The performance of the Decision Tree classification was, however, adequate but did not correspond to the accuracy of the classification levels for the ensemble.

Various study groups have produced several information maps launched in 2012, for example Google KG [28]. Search engines were used for returning links for the queries requested before 2012. When the Information Graph was presented by Google (KG), the scenario has been changed. This graph is nothing more than the information base used to find and address questions in the Google search engine. The data obtained from the knowledge graph are presented in the information box (which is displayed on the right side of the web page besides links). The Chatbox is also recognized as an information desk. Google recently began using the Information Map to answer Google Home spoken questions. The graph is written in English, French, Spanish, Spanish, German, etc. Critics earn Google's Information Graph.

Many literature works suggest that ontology be constructed automatically from textual resources using methods and tools for ontology learning. Most of the works linked to it are in the single sector, and most of them are in the political field. The principle of extraction was mainly based on the writers. Some researchers employed the methodology of reengineering to reuse ontologies. Furthermore, the authors suggested a reengineering approach focused on reusing ontology online. These findings prompted us to concentrate on developing a particular semantic model that can maintain knowledge relevant to the matching necessity. Firstly, multiple domain applications are different from our work. Second, it can extract all key elements in a domain-specific ontology from the proposed machine learning model. Finally, we plan to construct an expressive domain ontology that can correct the mistakes, update new terms and add relationships and axioms to the derived ontology in order to make them more explicit.

The organization of the rest of the section in the paper is as follows; section 3 describes the methodology used in the study in order to get the automated ontology. Section 4 shows the results obtained from the experiments that find the appropriate approach in order to develop ontology automation. Section 5 discusses the current study and future scope, followed by the conclusion.

3. METHODOLOGY

The proposed model in this study is based on drawing of the Jaccard relationships from text documents and ontological modeling of concepts and relationships. A combination of two different exclusison methods is the proposed innovation in the scientific model: the semantical and the thematic diagrams. A third method analyzes external service descriptors to validate the findings.

The three approaches point to the viability of the model. Additional advanced techniques, such as Machine Learning and Information Recovery (EIS), can also apply the model. However, the use of direct methods underlines that certain methods can be" "recounted for"," and the findings are attributed to the mixing and verification process of the model. Figure 1 shows the overall Ontology process.

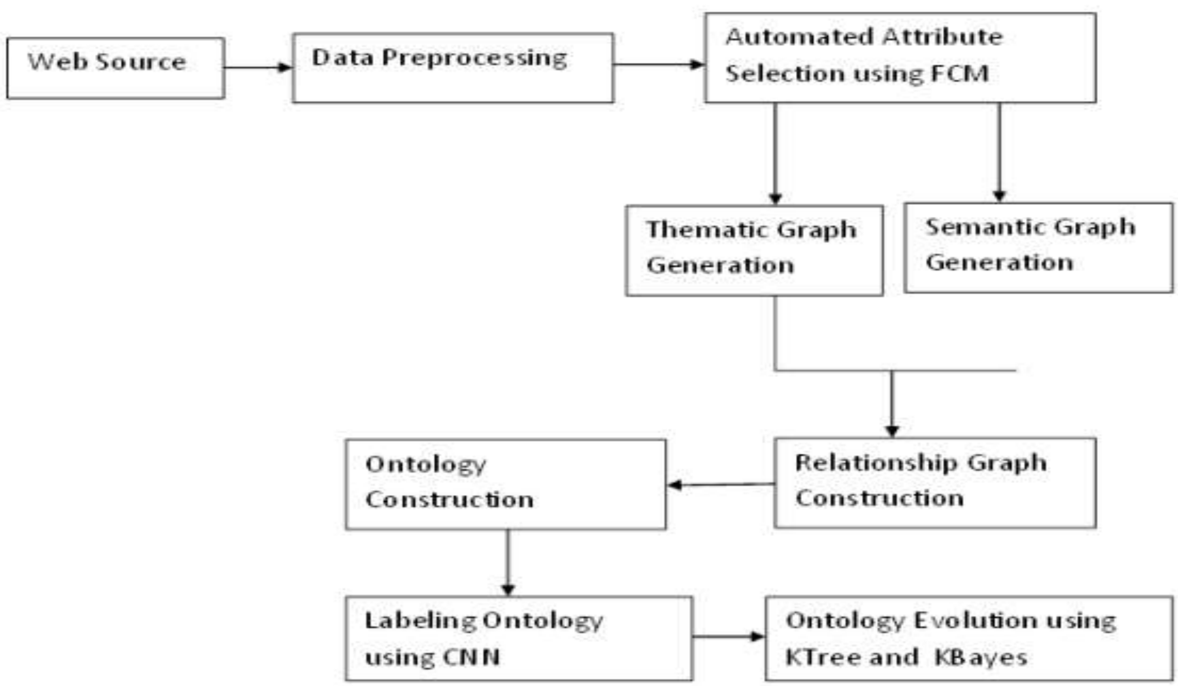


Figure 1: Ontology Designing Process

The proposed approach employs four type of algorithms to automatically acquire information, extract the attributes, connect the entities, and validate the entity. Data gathered from different sources is first stored in its native format in the data lake before they are sent, in the form of tables, to the IKA (Improved Kidney-Inspired Algorithm). The data lakes are the repository of raw information available in data warehouses, which is a data science discipline about enormous data study. . A data lake is a place where data can be obtained and used for analysis in an unprocessed format. The IKA algorithm uses three processing techniques: normalization, harmonization, and construction of a decision tree. The graphical data tables are logically arranged, and redundant information is extracted during normalization. The graphical data tables are then submitted to the harmonization process once the normalization process is completed. The possible combination of data from heterogeneous sources is generated during harmonization. After the pre-processing steps have been completed, the graphic databases are translated to ".arff" file format to insert information into ontology. The decision tree is based on its principles and attributes.

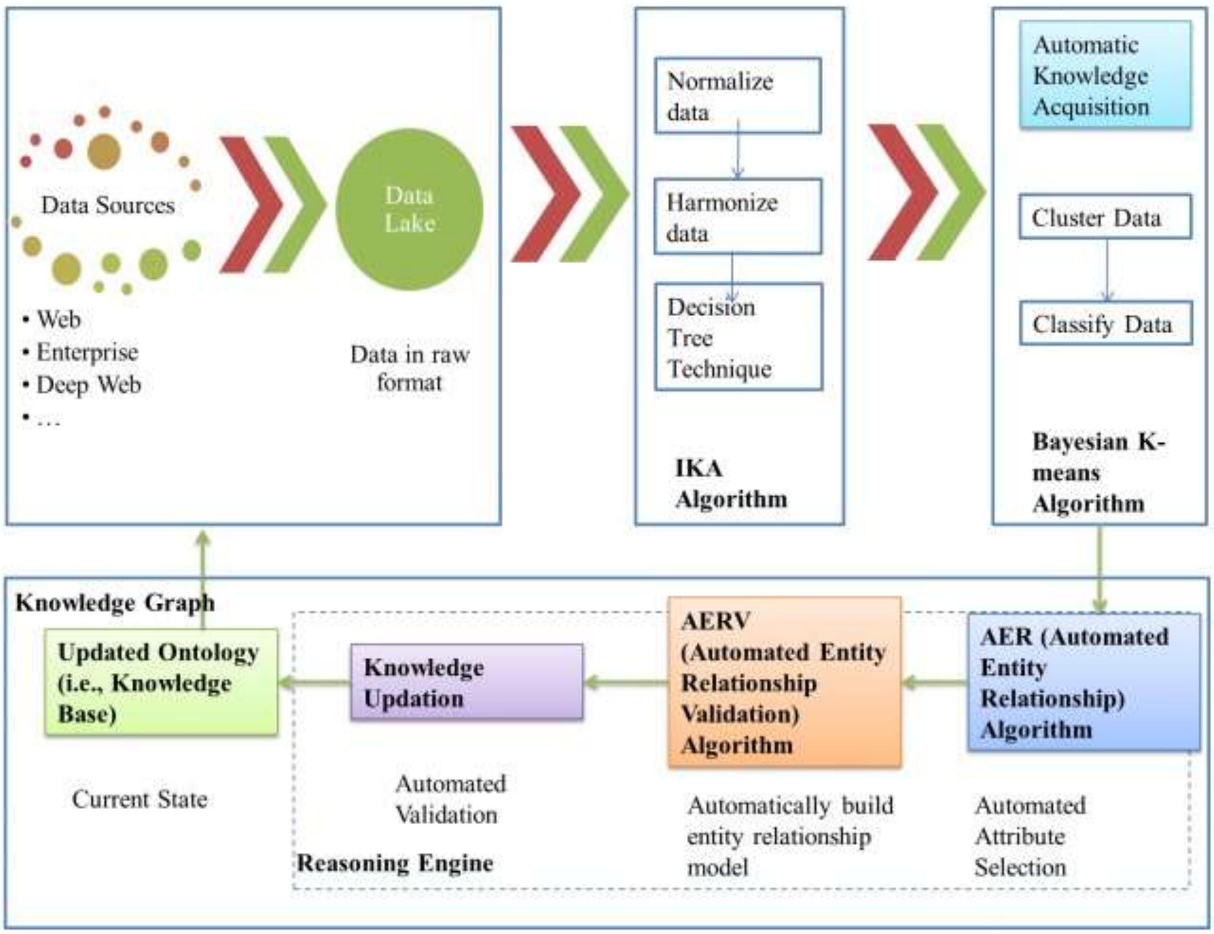


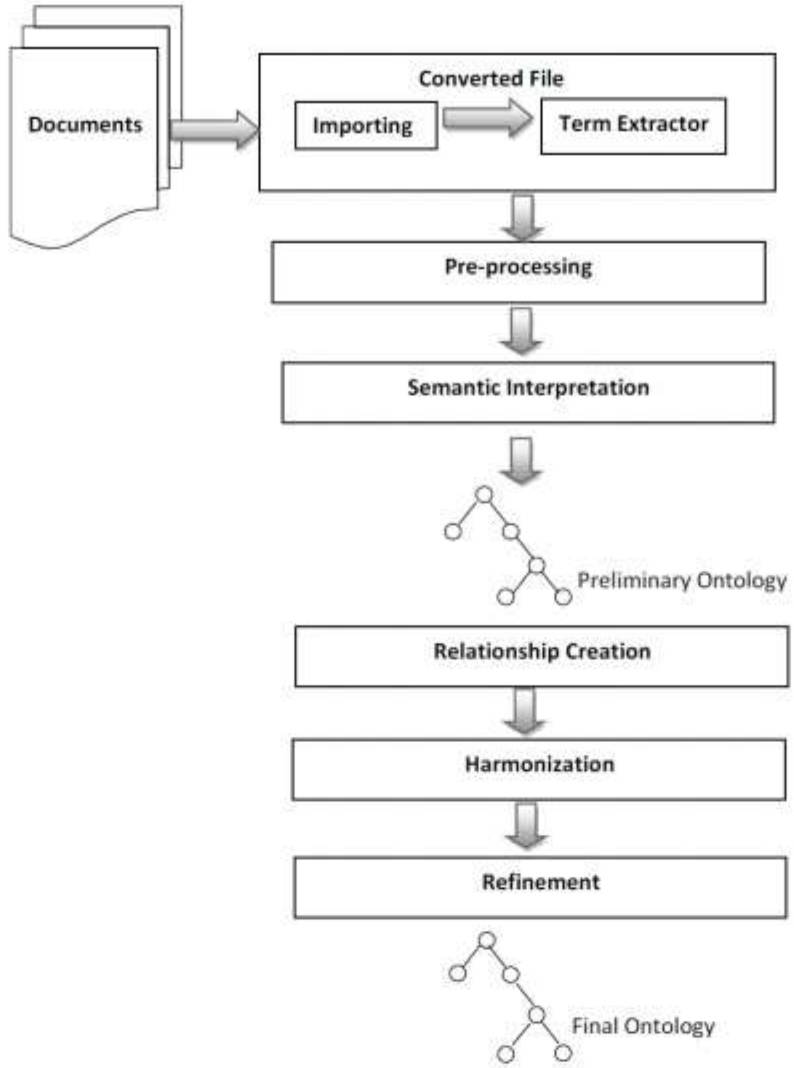
Figure 2: Proposed framework

Furthermore, the decision tree obtained will be sent to the K-means Bayesian algorithm, which clusters the decision tree and performs the classification. The clustering of the decision tree is initially based on the parent 'nodes' average value. The clustered data shall then be transmitted to the Naïve Bayes algorithm for classification. The proposed algorithm Naïve Bayes is chosen to cluster data since it provides a higher precision than other current algorithms. The categorized result indicates the best possible classification of the ontological attributes.

The classified characteristics are then injected into the Automated Entity Relationship Algorithm (AER). The algorithm obtained from the relation of the entity based on the attributes of the category. The AER algorithm is used to combine the relationship of an entity with the classified attribute through mapping techniques. Finally, the entity relation model obtained is transmitted to the Validation Automated Entity Relationship Algorithm (AERV). The AERV investigates the infringement of modeling rules by the model received. Further, it verifies that the syntax is true, that the models obey positions, and that the conditions of the statement are met. For more awareness updates, the proposed validation technology is highly important. Finally, the result is converted into an accurate ontology. Figure 2 demonstrates the proposed structure.

Intelligent Knowledge Acquisition (IKA) Algorithm

1. Creates an N node.
2. If whole T records have the same goal category
3. Back N as a goal community leaf node.
4. If there are empty attributes available.
5 Return N as the overall goal community leaf node for the records.
6. Obtain the best attribute (T, available attributes).
7. Attributes available = best attribute attributes available.
8. Divide the records according to best attribute (best attribute, T)
9. Ti of T on the best attribute for every break.
10. Append the IKA Decision Tree (split Ti records, attributes available) to a new node returned
11. End loop
12. End function



.

Figure: 3 Extraction of Domain-Specific Ontology

Step	Description
Pre-processing	The records can be retrieved here. This phase is comprised of many sub-phases, listed below. 1) Converting formats: Migration of documents to a more suitable (say, XML) occurs. 2) Stemming: here, the words in the document examined are reduced by a combination of different algorithms to their root form. 3) Marking speech components: Words (also multi-word terms) in a text that corresponds to a particular voice element are here marked in the document (e.g. nouns, adjectives, verbs, etc.). 4) Avoiding word listing: here, excessive domain requirements are deleted (e.g. conjunctions, articles, and verbs).

	5) Synonym recognition and terminology extraction
Establishing the Ontology	A basic version of ontology is developed based on primitive terms with simple and compound concepts.
Mapping of Relationship and Concept	Different statistical and ML algorithms are used in data mining to identify concepts and relationships in the ontology generated. Three major types of ML algorithms are available: un-attention monitored and semi-controlled.
Harmonizing	This is considered optional and appropriate if a consumer wants to harmonize the ontology obtained with the available knowledge base. To expand the knowledge base available, two or more ontologies are combined into a single ontology.
Validating and Refining	The objective ontology is tuned here and supported by its changing existence. To evolve the particular application and also its ongoing growth, the adaptation, and refinement of ontology, taking into account user requirements, play a crucial role. Cutting out the extracted ontology of unrelated definitions is a significant step.

3.1 Pre-processing

The pre-processing starts the whole process. The data is converted into a data cleaning phase during the pre-processing. Text is just a series of sentences, or rather a sequence of characters. However, in the field of language modeling or linguistic processing, we are not focused on the depth of our ’data’s character, but the whole sentence. One explanation is that the various characters do not have a broad history in the language model. Characters such as "“d”,” “r,” "a" or "e" have no significance, but they can form "readings" when rearranged into sentences, which can clarify those acts which probably have already been done just that.

3.1.1 Vectorisation

A vector is a method of giving the computer information using numbers. The data collection and data collection methods can vary from study to study.

3.1.2 Unwanted characters elimination

The process of clarification of the text is the key step. If a text is not included in the HTML/XML source, it should delete all non-alphabetical dots, other non-language character types, and HTML classes. Popular cleanup methods include regular expressions to filter unwanted messages. Many systems have key characters in English, including stop signs, questions, and unpredictable symbols.

3.1.3 Tokenization

Tokenisation is just the way an expression is broken into phrases.

3.1.4 Removal Stop-words

The two common termination removal methods are available, both of them simple. One approach is to count and assign a number value to each of the event words and delete any words/words that are more than the given value. Another option is to provide a predefined breaking set of symbols/symbols, which can be omitted from the list. In systems that rely on sentimental/sentimental analysis, such as "Lola Lov," some people can use helpful information “Blues Basset” except for the systems that need a more structured type of application. This can be done as well.

3.2 Graph selection attributes and relationship building

The next stage is to choose an attribute from the pre-processed data and apply it to the target data. We have minimized the data. To reduce the amount of data, the orignal information has been optimized. Word order in the attribute must be well chosen to avoid redundancies.

3.2.1 Fuzzy C-Means Clustering attribute collection

It involves the division by weight values of pre-processed data into different groups. Therefore in the same class, this attribute is as similar as possible. Since clusters may be viewed as a set of group data collections, clustering methods based on the flexibility or transparency of the subsets can be used. Fluorescent tightening device performance is typically greater than other clusters of the current one. The Fuzzy-C implies that the input data of each cluster is placed in a group of clones at a certain point.

The (Fuzzy C-means) FCM's main concept is to reflect the similarity between a cluster point. FCM executes membership through an affiliate function that differs in all clusters from zero to one at any sampling stage. Depending on the environment of the cluster, the cluster may be small, medium, or large. For any sampling point, the number of population must be the same. This is the summary of the FCM Clustering Algorithm.

Let P={p1, p3,….. pq} be a data collection, where each pq data point is a size n vector. Uvq, set of real matrices, and v-integer, two separate matrices. Then the C-mean space for the P partition is,

$$MF_{FCM} = \{U \in Uvq: U_{jk} \in [0,1]\} \quad (3.1)$$

$$\sum \mu jk \ c \ j=1 = 1 \text{ where; } 0 \leq \mu jk \leq 1,\ k = 1,\ q\ \} \quad (3.2)$$

μ_{jk} of the membership of kth data point in the jth cluster, $j = \{1,2,3, \dots \dots c\}$ (3.3)

The algorithm begins with random selections of centers, then the Fuzzy membership for each attribute is calculated in every iteration until the cluster centers are not changed. The attributes are given to the group with the highest number of members. The method is optimized to work with the unique set of parameters (3.4)

$$H_m = \sum_{k=1}^{n} \sum_{j=1}^{c} \mu_{jk}^{m} d^2 \ (p_k v_j) \qquad (3.4)$$

Where

$$V_j = \frac{\sum_{j=1}^{n} \mu_{jk}^{m} p_k}{\sum_{j=1}^{n} \mu_{jk}^{m}}$$

$$\mu_{jk} = [\sum_{i=1}^{c} [\frac{d(p_k v_{j)}}{d(p_k v_i)} \wedge \frac{2}{m-1}]]\wedge^{-1}$$

Equation (3.4) is a minimum square feature, with parameter n setting and the number of groups under which data is grouped by parameter c.. d^2 ($p_k v_j$) is Euclidean, v_j is the center vector of the j^{th} cluster and p_k is the vector of the k^{th} attribute. When a time limit or a preset number of iterations is reached a small positive constant of 0 to 1, μ is the termination criterion, . this method is stopped. H_m is the aim of assigning each cluster attribute.

FCM Steps:

1. Choose the cluster number.

2. Offer a certain point in the cluster the feature

3. Repeat Steps 2 and 3 after the algorithm concludes. Using equation (3.5) above to determine the insulin for each cluster.

4. Solve the member's class equation: (3.4)

This algorithm reduces the variance of clusters but suffers from the same issues as the k-means clustering algorithm. Choosing a number for a minimum local level will depend on the initial weight selection. A gaussian mix is used to test class membership with data. Another factor closely related to the fuselage is Kloss. These things are useful data processing tools for grouping objects. Mathematicians introduced the "delay" into the FCM algorithm to reduce noise consistency problems.

Construction of Semantic graph

The first step in preparing an ontology-based classification is to create a raw graph from the raw data set. Graphics and the symbolic relationship between phrases in the text must be removed from the document. The ontology topic is listed in a file with the matching clause for the subject consonants (used as the subject name). These words are also interpreted as values for certain attributes in the formulation value and used as their identity. We probably suppose that the name of an Entity is described by these characteristics (usually known as a label) and that they decide the same meaning (nickname). The unit weight is based on the weight measured utilizing markers. We find label matches to be a quality thing. The topic names can be linked in the text in many different places.Simmilarly , experts in advanced disciplines such as linguistics, are required for textual analysis. The randomness of the unit is tied to several physical phenomena. We measure the cumulative weight gain of individuals via the following formula:

$$\mathrm{w} = \frac{1}{1+\sum_{t=1\ldots n} s_{i*} p_i} \tag{3.5}$$

There are two steps in calculating the new weight: Pi is used to compare the similarity between any two sets of words.

Thematic graph

Analytical terms can address more than one subject. Many items may also, in random phases, even if they are not related to or perhaps associated with the document's principal subject, be incorporated into ontology graphics. Moreover, specific provisions in the file may lead to several identifiers but maybe the only one to reflect the right match in the context of a document. The sequence of this algorithm is to choose the best definition of the unit and the recognized relationship from the diagram.

The choice of the graph is based on the assumption that the components are related to each other and constitute a graph. Graphs are made using people and theological ties, so the subject and relations in this part must be one subject (category). A topic in a graph not connected with or belonging to another element, maybe a smaller group of similar components, most of which belong to another subject. Where the document focuses on an automated texting classifier topic, a single or minimal dominant thematic chart that corresponds with the main subject of the document should be included in the digital chart of the file. We have selected thematic graphs with the most artefacts and the most elements for further study and classification. If there were similar figures in other thematic maps, they would be included for further study. If one choose more than one thematic graph, it can show that the file deals on more than one topic. In this way, dominant graph selection effectively excludes elements that are not the main subject of research. Besides, the decrease in graphite means that porous (or slotted) parts are removed which are poorly associated with the overall picture. This is the step of minimizing low-cost data and restricting the flow of troubling data. Calculation of the center points for a subject, normally the subject mark, in the thematic map. Most central units were located at a geographical scale in our studies. The sum of the shortest pathways in the composition between the selected pitches and other pitches determines a geographical location measurement:

$$X_k = \frac{1}{1+\sum_t S(X_{k,}X_L)} \tag{3.6}$$

Wheres(Xk, Xl) is between Xk and Xl the shortest path width. The best institutions and core organizations, as a pillar of the thematic graph. Their importance to the subject of the document is decided. Notice that the most central and vice versa should not be the strongest authorities.

3.3 Semantic clustering

Semantic features are added in order to incorporate the semantic dimension in the clustering process. Because semantic value is translated into conception (i.e. corresponds to definition marks in the ontology of reference [28]. Instead of basic modalities is Comparisons between values can be made between Using a feature of semantic resemblance.

For comparing objects in a semantic, the concept of distance/similarity measurements between the values of a semantic pair is critical approach to cluster. The similarity is quantified by how concepts are like in some information (e.g. ontology or a corpus) based on semantical proof. The knowledge used to approximate the similarity between words enables these functions to be categorized in various families. In some approaches, taxonomies and ontologies in general are regarded as a graphic model that models semantic relations as a connection between concepts. The similarity then typically depends on the minimum number of connections between concepts (that is the minimum path). Similarity may also rely on other features like the breadth of the taxonomy concepts. This taxonomy measure has the key advantage of depending only on ontology to determine the similarity. However, the degree of completeness, homogeneity and coverage of ontology affect them.

4 RESULTS AND DISCUSSION

4.1 Experimental Setup

The experiment of this study has examined ontology from various domains. Agriculture, cancer, pizza, books, and banking have been chosen to create a multidisciplinary science. Besides, the vast amounts of unstructured text, robots are searching for terms. This teaching method is a compilation of the University of Farming and the Cancer Institute and Pizza and the Central Bibliography of more than 68,000 papers. The paper contains a total of 47,600 documents, of which 13,600 are research articles and 6,800 are test articles.

4.2 Efficiency Parameters

Several performance metrics are available to measure the efficiency of the proposed ontology-building process. To evaluate performance, this paper uses the detection accuracy, accuracy rate, and Specificity, Precision Rate, Recall Rate, F-Measure, and error rate.

Experiment No #1: Performance Analysis of CNN

In this experiment, the proposed CNN solution will be tested by adjusting the dimensions of the layers. The kernel size of this experiment is 3 to 11. The output values are calculated by adjusting the layer size. The performance analysis of CNN in different layer sizes is summarised in Tab. 1.

The findings from Tab. 1 indicate that CNN has the highest accuracy in 11th layer classification. This work also takes into account the difficulty of time for the best layer size range. Even if the 11th layer yields more results than the 7th layer. The time of its execution reaches the 7th layer and is 9800 seconds. (It takes 8800 to perform). Layer size 11 however, produces only minor fractional changes. Thus, the best result is 7 layer scale.

Tab. 1: Performance analysis of CNN on Various Layer Sizes

Performance Metrics	**Layer Size**				
	3	**5**	**7**	**9**	**11**
Accuracy	0.965	0.967	0.969	0.972	0.974
Error	0.035	0.033	0.031	0.028	0.026
Precision	0.975	0.977	0.979	0.982	0.984
Recall	0.961	0.962	0.963	0.970	0.972
Sensitivity	0.985	0.987	0.989	0.992	0.994
Specificity	0.972	0.974	0.976	0.981	0.982
F-Measure	0.942	0.944	0.946	0.949	0.951

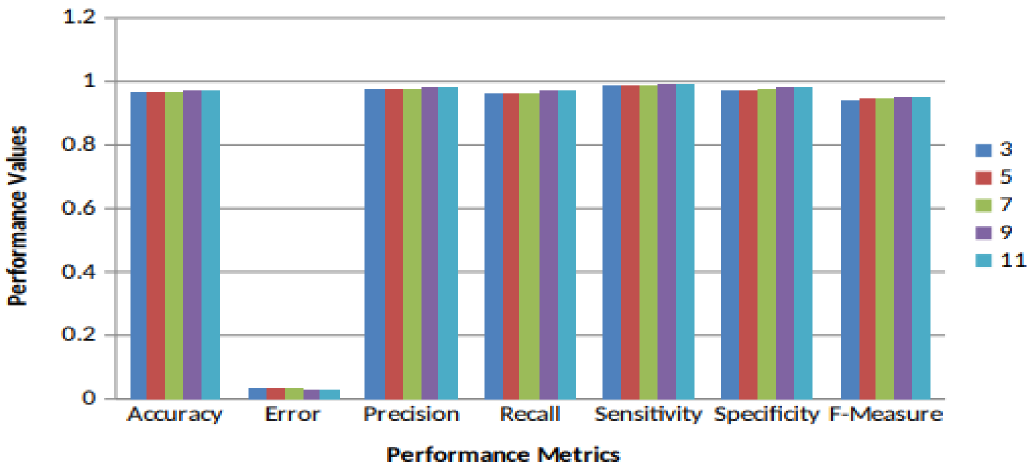


Figure 4: Performance analysis of CNN on Various Layer Sizes

Experiment No #2: Performance Analysis of FCM Segmentation Approach on Various Termination Iteration Count

In this experiment, the FCM segmentation approach will be tested by adjusting the termination iteration number. The kernel size of this experiment is 100-1500. The detection accuracy values are calculated by adjusting the termination iteration. The performance analysis of the FCM in different termination numbers is shown in Tab. 2.

Tab. 2: Performance analysis of FCM in Various Termination Iteration Count

Performance Metrics	Termination Iteration Count				
	100	300	500	1000	1500
Precision	0.8971	0.9091	0.9201	0.9301	0.9421
Accuracy	0.8871	0.8991	0.9101	0.9201	0.9321
Error	0.113	0.101	0.09	0.08	0.068
Recall	0.8771	0.8891	0.9001	0.9101	0.9221
Specificity	0.9171	0.9291	0.9401	0.9501	0.9621
Sensitivity	0.9271	0.9391	0.9501	0.9601	0.9721
F-Measure	0.8941	0.9061	0.9171	0.9271	0.9391

Tab. 2 reveals that the FCM segmentation method in the 1500th iteration provides the highest rating accuracy. This work considers the higher precision value for the best number of termination iterations. A higher result than an iteration count is obtained with the 1500th iteration. The best result is therefore the 1500th ending iteration count. Figure 5 displays the diagram of the effective detection analysis.

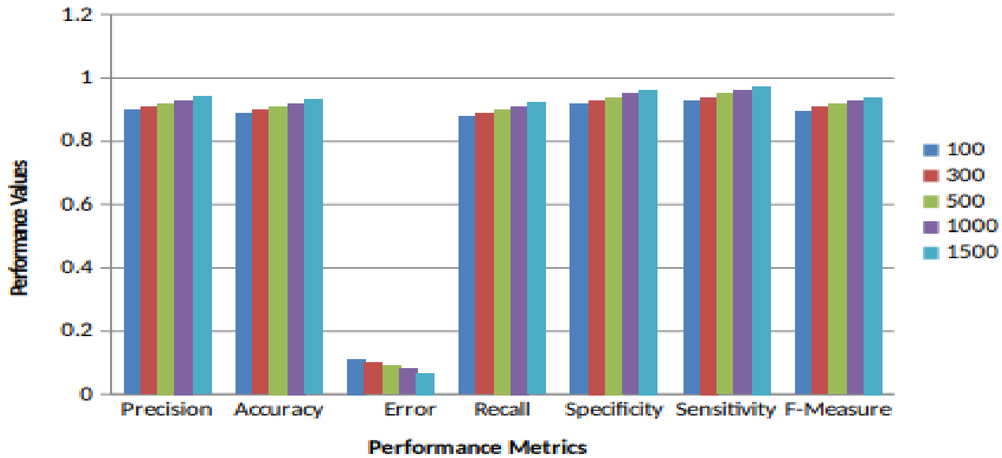


Figure 5: Performance of FCM Method on Various Termination Iteration Count

Experiment No #3: Performance Analysis of FCM Segmentation Approach on Various

Cluster Sizes this work assesses the FCM by adjusting the cluster size in this experiment. The cluster size of this experiment is 10-100. The output values are calculated by adjusting the cluster size. The performance analysis of the FCM in different cluster sizes is shown in Tab. 3.

Tab. 3 shows that the FCM segmentation method achieves the maximum cluster accuracy of 100. This work considers the higher accuracy value for the best selection of cluster size. The scale of group 100 is larger than other sizes in the colony. This means the optimal result is the cluster size of 100. Figure 6 displays the graph of the detection exactness analysis

Tab. 3: Performance analysis of FCM in Various Cluster size

Performance Metrics	Cluster Size				
	10	30	50	70	100
Precision	0.9021	0.9051	0.9101	0.9191	0.9301
Accuracy	0.8921	0.8951	0.9001	0.9091	0.9201
Error	0.108	0.105	0.1	0.091	0.08
Recall	0.8821	0.8851	0.8901	0.8991	0.9101
Sensitivity	0.9221	0.9251	0.9301	0.9391	0.9501
Specificity	0.9321	0.9351	0.9401	0.9491	0.9601
F-Measure	0.8991	0.9021	0.9071	0.9161	0.9271

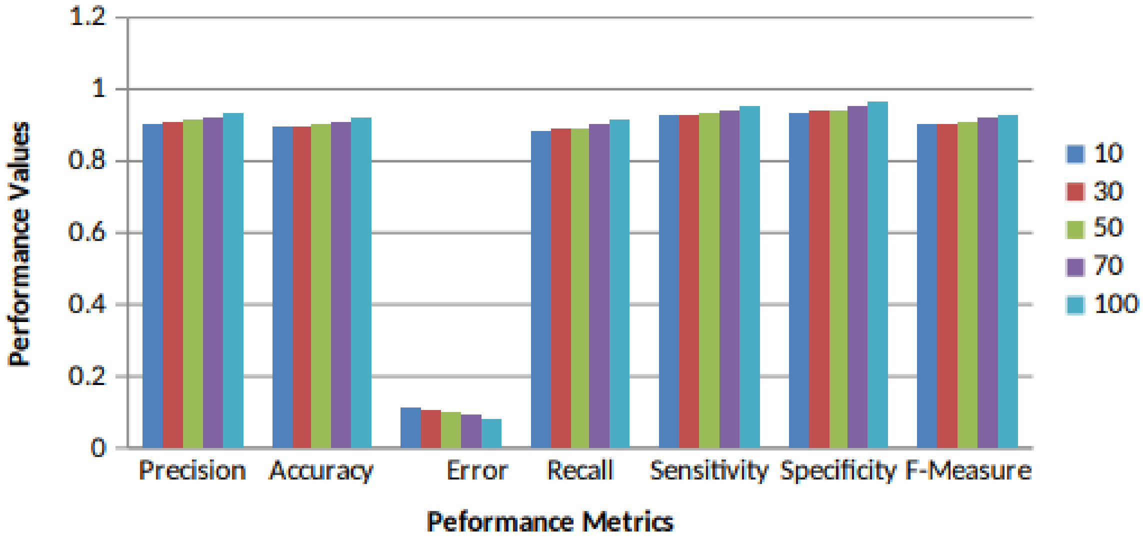


Figure 6: Performance of FCM Method on Various Cluster Size

Graph 6 shows that as the cluster size is increased, the detection accuracy value of the FCM system is to increase. The cluster size of this experiment is 10 to 100. The error rate values are calculated by adjusting the cluster size.

4.3 Ontology Evolution

Pathology has been developed in four stages:

1) To generate new concepts,

2) Defining the relation

3) Interaction forms recognition.

4) Restart the next WSDL file setup process.

Emerging theories can instead be evaluated for existing pathologies. The K-Tree and -algorithms are used to determine the relationship between concepts.

It uses the Jaccard relationship estimator to create the Ontology Model. The next step is to designate the built-in ontology based on the new network after the algorithm has been built. The ontological architecture based upon the CNN is divided into two stages, e.g. offline and online. Application labels based on different domains classify the number of data into different groups. Pre-processing, selection, assignment, and theorem formation with a loss function were conducted in the study phase to build predictive models. Next, mark the data set for the preparation. Data Size Before adjusting the size of the data. Finally, network neural channels are used to create pathology automatically. The data set has been deleted from the net. The application is one of the models that have previously been established through demonstration. If training is chosen to train from the very first layer, the entire layer (meaning) must be trained to the final layer. The consumption of time is therefore very high. The efficiency is affected.

A brain-training model was used for the classification phase to prevent this problem. The loss function is determined by using the algorithm of gradient generation. Classification using the evaluation feature was compared to raw image data. The efficiency of a particular collection is calculated according to the loss function. The results are based on the key marks in the accepted offline data. To improve accuracy, measuring the loss function is important. The accuracy is poor when the loss feature is very high. Likewise, the precise and low loss function is high. The value of the pitch is determined to determine the pitch algorithm as a loss function.

To measure the loss gradient function, repeatedly assess the gradient value. The algorithm is seen in algorithm 2 for the CNN-based classification.

The second algorithm

1. Apply first layer convolution filter

2. Filter sensitivity is reduced by smoothing the convolution filter (i.e.

3. The activation layer 4 governs the transition of the signals from one layer to another.

4. Enhance the duration of training with the linear corrected unit (RELU)

5. Neurons are bound to any neuron in the following layer.

6. Add a loss layer to send the neural network feedback during an offline phase at the end

5 CONCLUSION

This research has successfully developed a complete domain of ongoing ontology using algorithms for master learning and text corpus in domains such as health, agriculture, and food. In this research work, the ontology is focused on the extraction of Jaccord relations from text documents and the use of conceptual and ontological models for relation. The model is correct by two separate extraction models, such as the automation ratio (AER) and the automation of subjective contact (AERV). A category of descriptions is categorized to set the meaning of a web service. The utility of the technique suggested is evaluated in different fields. The main aim of a full automated ontology build is to minimize the manual involvement for ontology updating and assessment by human experts. The architecture and framework described above to automatically generate a domain-specific ontology based on several advantages of existing machine learning algorithms. Four algorithms are the most fundamental in the different stages of ontology development. Secondly, the approach anticipates the perfect classification of ontology attributes. Thirdly, validation methods are suggested for the next updating processes of information.

A library of the trained and tested Ontology components can be created, which must include the lexical context of unstructured knowledge. In existing Ontology repositories, the proposed model can be checked. Multi-linguistics and multi-linguistic ontology can be built into the proposed process. The proposed model, which allows ontological constructors to create ontologies based on their domain requirements, may serve as an illustration for a steady Ontology model.

Availability of supporting data: The datasets generated during and/or analysed during the current study are available from the corresponding author on reasonable request.

Competing interests: The authors declare that they have no conflict of interest.

Funding: No funding was received for this manuscript.

Authors' contributions: The first author is a major contributor. The second author has given the inputs regarding CCN and reviewed the manuscript.

Acknowledgements: We would like to express our sincere gratitude to Dr. Manju Pandey for her continuous support and suggestions. We would also like to thank KL University (KLEF) for its constant cooperation and support.

REFERENCES

[1] A.T. Luu, Y. Tay, S.C. Hui and S.K. Ng, Learning term embeddings for taxonomic relation identification using dynamic weighting neural network, in: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, J. Su, K. Duh and X. Carreras, eds, Association for Computational Linguistics, Austin, Texas, 2016, pp. 403–413, https://www.aclweb.org/anthology/D16-1039. doi:10.18653/v1/D16-1039.

[2] G. Petrucci, M. Rospocher and C. Ghidini, Expressive ontology learning as neural machine translation, Journal of Web Semantics 52 (2018), 66–82. doi:10.1016/j.websem.2018.10.002.

[3] J. Völker, D. Fleischhacker and H. Stuckenschmidt, Automatic acquisition of class disjointness, Journal of Web Semantics 35(P2) (2015), 124–139. doi:10.1016/j.websem.2015.07.001.

[4] X. Yin, D. Gromann and S. Rudolph, Neural machine translating from natural language to SPARQL, CoRR 2019, http://arxiv.org/abs/1906.09302.

[5] V. Efthymiou, O. Hassanzadeh, M. Rodriguez-Muro and V. Christophides, Matching web tables with knowledge base entities: From entity lookups to entity embeddings, in: The Semantic Web – ISWC 2017, C. d'Amato, M. Fernandez, V. Tamma, F. Lecue, P. Cudré-Mauroux, J. Sequeda, C. Lange and J. Heflin, eds, Lecture Notes in Computer Science, Vol. 10587, Springer International Publishing, Vienna, Austria, 2017, pp. 260–277. ISBN 978-3-319-68288-4. doi:10.1007/978-3-319-68288-4_16.

[6] D. Gromann and T. Declerck, Comparing pretrained multilingual word embeddings on an ontology alignment task, in: Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018), N. Calzolari, K. Choukri, C. Cieri, T. Declerck, S. Goggi, K. Hasida, H. Isahara, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, S. Piperidis and T. Tokunaga, eds, European Language Resources Association (ELRA), Miyazaki, Japan, 2018, pp. 230–236, https://www.aclweb.org/anthology/L18-1034. ISBN 979-10-95546-00-9.

[7] E. Jiménez-Ruiz, A. Agibetov, M. Samwald and V. Cross, Breaking-down the ontology alignment task with a lexical index and neural embeddings, CoRR (2018), http://arxiv.org/abs/1805.12402.

[8] L. Qiu, J. Yu, Q. Pu and C. Xiang, Knowledge entity learning and representation for ontology matching based on deep neural networks, Cluster Computing 20(2) (2017), 969–977. doi:10.1007/s10586-017-0844-1.

[9] G. Burel, H. Saif, M. Fernandez and H. Alani, On semantics and deep learning for event detection in crisis situations, in: First Workshop on Semantic Deep Learning (SemDeep-1) at the European Semantic Web Conference, 2017, http://semdeep.iiia.csic.es/program.html.

[10] S. Shekarpour, F. Alshargi, K. Thirunaravan, V.L. Shalin and A. Sheth, CEVO: Comprehensive EVent ontology enhancing cognitive annotation on relations, in: 2019 IEEE 13th International Conference on Semantic Computing (ICSC), D. Bulterman, A. Kitazawa, D. Ostrowski, P. Sheu and J. Tsai, eds, IEEE, Newport Beach, CA, USA, 2019, pp. 385–391. doi:10.1109/ICOSC.2019.8665605.

[11] S. Thoma, A. Rettinger and F. Both, Towards holistic concept representations: Embedding relational knowledge, visual attributes, and distributional word semantics, in: The Semantic Web – ISWC 2017, C. d'Amato, M. Fernandez, V. Tamma, F. Lecue, P. Cudré-Mauroux, J. Sequeda, C. Lange and J. Heflin, eds, Lecture Notes on Computer Science, Vol. 10587, Springer International Publishing, Vienna, Austria, 2017, pp. 694–710. ISBN 978-3-319-68288-4. doi:10.1007/978-3-319-68288-4_41.

[12] J. Camacho-Collados, M.T. Pilehvar and R. Navigli, NASARI: A novel approach to a semantically-aware representation of items, in: Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, R. Mihalcea, J. Chai and A. Sarkar, eds, Association for Computational Linguistics, Denver, Colorado, 2015, pp. 567–577, https://www.aclweb.org/anthology/N15-1059. doi:10.3115/v1/N15-1059.

[13] M. Faruqui, J. Dodge, S.K. Jauhar, C. Dyer, E. Hovy and NA. Smith, Retrofitting word vectors to semantic lexicons, in: Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, R. Mihalcea, J. Chai and A. Sarkar, eds, Association for Computational Linguistics, Denver, Colorado, 2015, pp. 1606–1615, https://www.aclweb.org/anthology/N15-1184. doi:10.3115/v1/N15-1184.

[14] J. Goikoetxea, E. Agirre and A. Soroa, Single or multiple? Combining word representations independently learned from text and WordNet, in: Thirtieth AAAI Conference on Artificial Intelligence, D. Schuurmans and M. Wellman, eds, AAAI Press, Palo Alto, CA, USA, 2016, pp. 2608–2614. ISBN 978-1-57735-760-5.

[15] I. Iacobacci, M.T. Pilehvar and R. Navigli, SensEmbed: Learning sense embeddings for word and relational similarity, in: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), C. Zong and M. Strube, eds, Association for Computational Linguistics, Beijing, China, 2015, pp. 95–105, https://www.aclweb.org/anthology/P15-1010. doi:10.3115/v1/P15-1010.

[16] M.T. Pilehvar and N. Collier, De-conflated semantic representations, in: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, J. Su, K. Duh and X. Carreras, eds, Association for Computational Linguistics, Austin, Texas, 2016, pp. 1680–1690, https://www.aclweb.org/anthology/D16-1174. doi:10.18653/v1/D16-1174.

[17] L. Espinosa-Anke, J. Camacho-Collados, C. Delli Bovi and H. Saggion, Supervised distributional hypernym discovery via domain adaptation, in: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, J. Su, K. Duh and X. Carreras, eds, Association for Computational Linguistics, Austin, Texas, 2016, pp. 424–435, https://www.aclweb.org/anthology/D16-1041. doi:10.18653/v1/D16-1041.

[18] L. Espinosa-Anke, J. Camacho-Collados, S. Rodríguez-Fernández, H. Saggion and L. Wanner, Extending WordNet with fine-grained collocational information via supervised distributional learning, in: Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, Y. Matsumoto and R. Prasad, eds, Association for Computational Linguistics, Osaka, Japan, 2016, pp. 3422–3432, https://www.aclweb.org/anthology/C16-1323.

[19] J. Camacho-Collados, C.D. Bovi, A. Raganato and R. Navigli, A large-scale multilingual disambiguation of glosses, in: Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016), N. Calzolari, K. Choukri, T. Declerck, S. Goggi, M. Grobelnik, B. Maegaard, J. Mariani,

H. Mazo, A. Moreno, J. Odijk and S. Piperidis, eds, European Language Resources Association (ELRA), Portorož, Slovenia, 2016, pp. 1701–1708, https://www.aclweb.org/anthology/L16-1269.

[20] A. Raganato, C.D. Bovi and R. Navigli, Neural sequence learning models for word sense disambiguation, in: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, M. Palmer, R. Hwa and S. Riedel, eds, Association for Computational Linguistics, Copenhagen, Denmark, 2017, pp. 1156–1167, https://www.aclweb.org/anthology/D17-1120. doi:10.18653/v1/D17-1120.

[21] Hazman, M., El-Beltagy, S. R., & Rafea, A. (2011). A survey of ontology learning approaches. *International Journal of Computer Applications*, *22*(9), 36-43.

[22] Vicient, C., Sánchez, D., & Moreno, A. (2013). An automatic approach for ontology-based feature extraction from heterogeneous textualresources. *Engineering Applications of Artificial Intelligence*, *26*(3), 1092-1106.

[23] Fraga, A. L., Vegetti, M., & Leone, H. P. (2020). Ontology-based solutions for Interoperability among Product Lifecycle Management Systems: A Systematic Literature Review. *Journal of Industrial Information Integration*, 100176.

[24] Stenetorp, P., Pyysalo, S., Topić, G., Ohta, T., Ananiadou, S., & Tsujii, J. I. (2012, April). BRAT: a web-based tool for NLP-assisted text annotation. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics* (pp. 102-107).

[25] del Carmen Legaz-García, M., Miñarro-Giménez, J. A., Menárguez-Tortosa, M., & Fernández-Breis, J. T. (2016). Generation of open biomedical datasets through ontology-driven transformation and integration processes. *Journal of biomedical semantics*, *7*(1), 1-17.

[26] Safdar, S., Zafar, S., Zafar, N., & Khan, N. F. (2018). Machine learning based decision support systems (DSS) for heart disease diagnosis: a review. *Artificial Intelligence Review*, *50*(4), 597-623.

[27] Lagos-Ortiz, K., Medina-Moreira, J., Paredes-Valverde, M. A., Espinoza-Morán, W., & Valencia-García, R. (2017). An ontology-based decision support system for the diagnosis of plant diseases. *Journal of Information Technology Research (JITR)*, *10*(4), 42-55.

[28] Studer, R., Benjamins, V. R., & Fensel, D. (1998). Knowledge engineering: Principles and methods. *Data & Knowledge Engineering*, *25*(1–2), 161–197. https://doi.org/10.1016/S0169-023X(97)00056-6