Available at www.sciencedirect.com

INFORMATION PROCESSING IN AGRICULTURE 5 (2018) 60–73

journal homepage: www.elsevier.com/locate/inpa

Automatic relationship extraction from agricultural text for ontology construction

Neha Kaushik *, *Niladri Chatterjee*

Indian Institute of Technology Delhi, Hauz Khas, New Delhi 110016, India

ARTICLE INFO

Article history:
Received 13 February 2017
Received in revised form
18 November 2017
Accepted 21 November 2017
Available online 2 December 2017

Keywords:
Relation extraction
Term EXtraction
NLP
Ontology
Knowledge-based relation extraction
Self-supervised relation extraction

ABSTRACT

In the present era of Big Data the demand for developing efficient information processing techniques for different applications is expanding steadily. One such possible application is automatic creation of ontology. Such an ontology is often found to be helpful for answering queries for the underlying domain. The present work proposes a scheme for designing an ontology for agriculture domain. The proposed scheme works in two steps. In the first step it uses domain-dependent regular expressions and natural language processing techniques for automatic extraction of vocabulary pertaining to agriculture domain. In the second step semantic relationships between the extracted terms and phrases are identified. A rule-based reasoning algorithm RelExOnt has been proposed for the said task. Human evaluation of the term extraction output yields precision and recall of 75.7% and 60%, respectively. The relation extraction algorithm, RelExOnt performs well with an average precision of 86.89%.

1. Introduction

Development of domain specific ontology is one of the fastest growing techniques for knowledge representation, and its subsequent utilization. The domain of agriculture is no exception. Huge amount of agricultural data is available in the form of textual documents, tables and spreadsheets. However, the data is often underutilized because of lack of application of modern data processing techniques to it. In developing countries, such as India, the decision making is still primarily based on human experts and governmental policies. Factual corroboration with the help of existing data is still missing from the overall policy making. The present paper aims at bridging the gap. Our primary focus is to extract terms and their relationship from the existing texts using minimal domain knowledge towards creating an ontology [1] for agriculture domain with a focus on the Indian context.

An efficient algorithm, called RENT, for automatic term extraction in agriculture domain has already been proposed in [2]. The RENT algorithm is based on regular expressions and natural language processing techniques. In the present work we extend the scheme given in above-mentioned work further for automatic extraction of semantic relationships among the terms from the agriculture domain text to facilitate automatic creation of ontology. In particular we have developed and experimented with two different approaches, namely:

* Corresponding author.
E-mail addresses: swami.neha@gmail.com (N. Kaushik), niladri.chatterjee@maths.iitd.ac.in (N. Chatterjee).
Peer review under responsibility of China Agricultural University.
https://doi.org/10.1016/j.inpa.2017.11.003

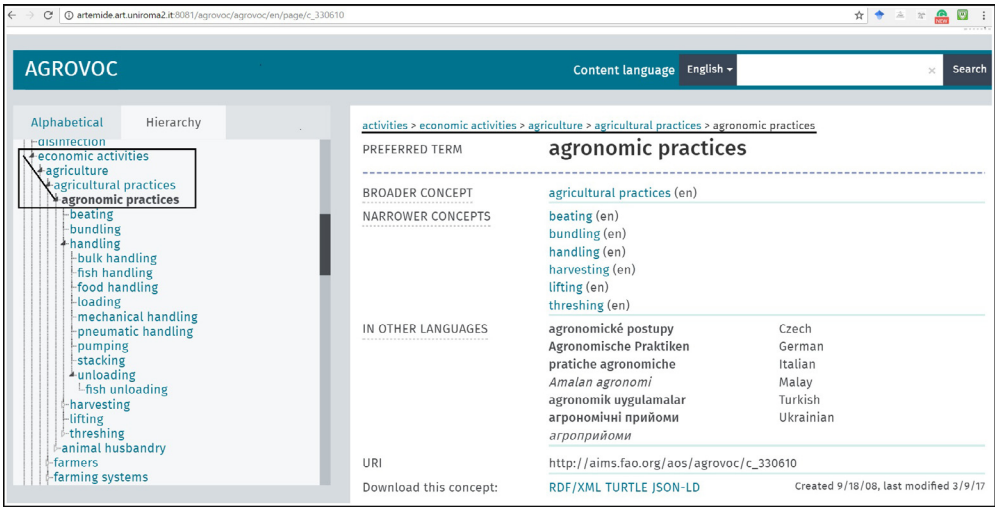

Fig. 1 – Screenshot from AGROVOC showing the hierarchical structure.

i. modified Open Information Extraction (mOIE),
ii. RelExOnt (Relation Extraction for Ontology) scheme

and studied their relative utility.

The motivation behind developing the mOIE scheme is from the well-known Open Information Extraction (OIE) [3–5] approach. However, we have modified the original scheme to suit the needs of the agriculture domain. The aim of mOIE is on identifying related terms pairs from a given list of terms and domain text without human intervention. RelExOnt, on the other hand, takes as input the domain text, a set of terms along with a list of relations as identified by domain experts. The scheme then extracts the related pairs of terms from the text satisfying each of the given relations.

Although several agricultural thesauri, such as NAL thesaurus,[1] AGROVOC,[2] are available online, they have certain deficiencies. For example, NAL thesaurus is too scientific for the actual users of the domain, *viz.* the actual cultivators who are often familiar with colloquial terms and not the scientific terms used in NAL thesaurus. AGROVOC, on the other hand, despite having a larger vocabulary in several languages, suffers from two major problems in the context of Indian Agriculture:

- Some important terms from Indian Agriculture domain are not present in AGROVOC. Examples include `alfalfa`, `mesta`, `nigerseed`, `urad bean`, `ladyfinger`, `masur`, `lotus stem`, `kharif crop`, `rabi crop`, `scallion`, `ridge gourd`[3] among others.
- Many terms present in AGROVOC, e.g., `play`, `activities`, `bodies`, `housewives`, `collections`, `students`, `teachers`, seem to be irrelevant from agricultural perspective.

Another major problem with the above resources is that they are typically organized hierarchically. Fig. 1 shows the hierarchical organization of AGROVOC. It is a screenshot from AGROVOC showing the hierarchy for '`agronomic practices`'. However, domain-specific relations in general do not possess hierarchical structure only, and agriculture domain is no exception. Some such agriculture-specific relations are:

is_intercrop: a relationship between two crops suggesting when grown along with each other produce a better yield.
grows_in_soil: a relationship between a crop and a soil suggesting whether the soil is preferable for the crop.
grows_in_weather: a relationship between a crop and a weather suggesting the suitability of the weather for the crop.

The present work is aimed at bridging this gap.

The paper is organized as follows. Section 2 presents a review of relation extraction schemes that have been proposed in literature in general context, and in the context of ontology creation, in particular, over the last decade or so. A brief overview of the RENT algorithm is given in Section 3. Sections 4 and 5 explain the mOIE and the RelExOnt algorithms respectively. Section 6 concludes the paper with possible future directions of work.

2. Related past work

Relation extraction as a part of ontology generation and ontology population (i.e. addition of new concepts to the ontology) has been pursued for more than a decade [6,7]. The task is challenging as different kinds of techniques are needed even for extraction of the same relationship from text. For illustration, consider extraction of `has_synonym` relation from the text shown in Fig. 2, taken from [8]. Occurrence of the word "`or`" between `brinjal` and `eggplant` allows one to identify the synonymous relationship between them. However, the same does not hold good between `aubergine` and `brinjal`, as this identification requires resolution of

[1] http://agclass.nal.usda.gov/.
[2] http://aims.fao.org/standards/agrovoc/functionalities/search.
[3] Last searched on 03/07/2ss017.

Brinjal or eggplant is an important crop of sub topics and tropics. The name brinjal is popular in Indian subcontinent and is derived from Arabic and Sanskrit whereas the name eggplant has been derived from the shape of the fruit of some varieties which are white and resemble in shape to chicken eggs. It is also called aubergine in Europe.

Fig. 2 – Example agricultural text [4].

the anaphoric pronoun "it", which cannot be captured through straightforward pattern matching.

As a consequence, relationship extraction received considerable attention in literature for last one decade. A comprehensive review of major relationship extraction schemes for ontology construction using Wikipedia can be found in [9]. Assiss and Casanova [10] present a relationship extraction technique using Wikipedia text and DBpedia ontology. A scheme for extracting patterns from a given news corpus text to identify substantial relationships using FreeBase[4] is presented in [11]. Lang and Lapata [12] present a scheme for Semantic Role Induction using several linguistic principles.

Relation extraction techniques, in general, are broadly classified into three categories [13]:

i. Knowledge based methods: These methods typically use patterns and rules crafted by human experts for extraction of relations from domain text. For example, some domain-independent patterns for extraction of hyponym relation are described in [14]. One major limitation of knowledge-based methods is that they are highly domain-specific, and hence their applicability in other domains is generally difficult, if not impossible. However, these methods perform effectively and achieve accurate results when input data is well defined.
ii. Supervised methods: These methods use machine learning techniques and training examples for relation extraction from domain text. Depending upon the techniques employed, several algorithms may be found under this category: bootstrapping methods (weekly supervised methods) [15], kernel methods [16], logistic regression methods [17], augmented parsing methods [18], conditional random fields [19] among others.
iii. Self-supervised methods: These methods are characterized by their ability to extract patterns for relation extraction automatically [20]. Some important self-supervised methods include Open Information Extraction (OIE), and distant learning [21]. OIE systems identify the sets of entities and textual patterns (probable relations) occurring between these entities in sentences from the domain text, while distant learning methods use some knowledge base for identifying patterns for extracting relations between entities occurring in the text.

The advantage of self-supervised models is that they do not rely upon domain or expert knowledge. One of the most recent self-supervised schemes found in literature, (cf. [22]) is explored for extraction of semantic and domain-specific relations from agricultural text. Table 1 shows the example text and output obtained using an online demo of the work given in [22].

Although the work presented in [22] identifies some meaningful terms from the domain text, it lacks in identifying domain-specific relations such as `is_intercrop` from the underlying text. This is evident from the text given at S. no. 1 in Table 1. Moreover, the related pairs of terms identified are not necessarily related by the corresponding relations identified by the scheme. For illustration, it identifies `food crops` as a sub-class of `integrated development`, which is not correct. Firstly, `integrated development` is not a valid agricultural term, and secondly, `food crop` is not a subclass of `integrated development`. In a similar vein, wheat identified as a subclass of integrated development is also not correct. Same argument holds for `Instanceof` (`fixes nitrogen`, `setaria`), `Instanceof` (`fixes nitrogen`, `rhodes`), `Instanceof` (`wide range`, `rhodes`).

This prompted us to design a novel self-supervised scheme for relation extraction from agricultural text. The proposed approach mOIE is a step towards this direction. However, our experiments with mOIE resulted in success only for identification of `has_synonym relation`. Therefore, we further develop the scheme RelExOnt, a knowledge-based scheme, where domain knowledge has been used for identification of other relations. In the present work we demonstrate the efficacy of RelExOnt for identification of `is_a`, `is_type_of`, and `is_intercrop` relations along with `has_synonym`, for which we have used a modified version of mOIE.

3. Automatic term extraction

3.1. *The RENT algorithm*

As mentioned in Section 1, the scheme given in RENT algorithm has been used for term extraction. The RENT algorithm uses domain-specific patterns in the form of regular expressions to extract single word terms as well as composite terms from agricultural text. Twenty such patterns have been used for identifying the initial list of candidate terms from texts on agriculture domain, as shown in Table 2.

These patterns have been selected by domain experts after careful analysis of more than 1000 pages of agricultural text,[5] collected from agriculture handbooks available from FAO,

[4] https://developers.google.com/freebase/.

[5] Since the agricultural text used in this work has been taken from various Government's websites and documents, we observed that their writing style tends to follow certain fixed textual patterns. The Regular Expressions have been formed by observing these patterns.

Table 1 – Example output for a recent self-supervised relation extraction scheme.

S. no.	Input text	Output terms	Output relations
1	The elephant foot yam is widely grown as intercrops in litchi, coconut, banana orchards	Intercrops, litchi, elephant foot yam, coconut, banana orchards	–
2	Integrated development of major food crops such as wheat, paddy, coarse cereals, minor millets, pulses, oilseeds	Minor millets, coarse cereals, oilseeds, pulses, paddy, food crops, wheat, integrated development	Subclassof (food crops, integrated development) Subclassof (wheat, integrated development)
3	Spices like black pepper, ginger, turmeric, vanilla, nutmeg, clove and some medicinal plants are the ideal intercrops for coconut	Spices, ginger, black pepper, vanilla, clove, medicinal plants, coconut, ideal intercrops, nutmeg, turmeric	Subclassof (spices, medicinal plants)
4	It fixes nitrogen very effectively and can be grown with a wide range of grasses such as Rhodes, Setaria, green panic and guinea grass	Setaria, guniea grass, fixes nitrogen, rhodes, grasses, green panic, wide range	Instanceof (fixes nitrogen, setaria) Instanceof (fixes nitrogen, rhodes) Instanceof (grasses, rhodes) Instanceof (wide range, rhodes)

Table 2 – Textual patterns and corresponding regular expressions for term extraction.

S. no.	Textual pattern	Corresponding regular expression
1	candidate-word season	(\w+),(season),2,word,$,$,$
2	candidate-word cultivation	(\w+),(cultivation),2,word,$,$,$
3	use of candidate-word	(use),(of),(w+),1,word,$,$
4	candidate-word systems	(\w+),(systems),2,word,$,$,$
5	consumption of candidate-word	(consumption),(of),(\w+),1,word,$,$
6	such as candidate-word	(such),(as),(\w+),1,word,$,$
7	production of candidate-word	(production),(of),(\w+),(),(w+),1,phrase
8	candidate-word hybrid	(\w+),(hybrid),2,word,$,$,$
9	growth in candidate-word	(growth),(in),(\w+),1,word,$,$
10	candidate-word crop	(\w+),(crop),2,word,$,$,$
11	cultivation of candidate-word	(cultivation),(of),(\w+),1,word,$,$
12	candidate-word production	(\w+),(production),2,word,$,$,$
13	candidate-word revolution	(\w+),(revolution),1,word,$,$
14	candidate-word sector	(\w+),(sector),2,word,$,$,$
15	including candidate-word	(including),(\w+),1,word,$,$
16	growth of candidate-word	(growth),(of),(\w+),1,word,$,$
17	millions of candidate-word	(millions),(of),(\w+),1,word,$,$
18	include candidate-word	(include),(\w+),1,word,$,$
19	candidate-word consumption	(\w+),(consumption),2,word,$,$,$
20	candidate-word productivity	(\w+),(productivity),2,word,$,$,$

Table 3 – Example of terms from list of candidate terms.

Valid terms	Score	Invalid terms	Score
Fodder	223	Private	38
Forage	212	Average	9
agriculture	121	Intensive	8
Oilseed	9	Dry	8
Brinjal	8	Appropriate	8
cultivation	7	Primary	7
intercropping	1	traditional	3

S.1 Identify the single words and bigrams occurring with these patterns and put them in a list, called Key_list.
- Remove any spaces before or after the extracted words in case of single words.
- Exclude the names of countries, states, city, and numbers whether in numerical or textual form.

S.2 Remove the stopwords from this list. The resultant list is named as Candidate_list.
S.3 In the Candidate_list, lets represent each word as W_i, i=1 to n.
S.3.1 For each W_i, assign a weight in the following way:
Weight(W_i)=1, initially, Weight(W_i)= Weight(W_i)+1, if W_i is a noun
S.3.2 For each pattern P_j in which W_i occurs, Weight(W_i)= Weight(W_i)+1
Weight(W_i)=a*weight(W_i)+b*frequency(W_i) //value of a and b to be decided experimentally
S.4 Sort Candidate_list as per weight(W_i), i=1 to n, in descending order.
S.5 Extract the words in the text that occur separated with commas. Let this list be C_list.
Remove stopwords from C_list. Append C_list to Candidate_list. We use the regular expression
'(?<=,)[^,\b\.]+'
S.6 Use linguistic filters to extract more terms
S.7 for each string CS extracted in S.6
if either of the word is in Candidate_list and length of CS<=15
append CS to Candidate_list

Fig. 3 – Overview of the RENT algorithm for automatic term extraction in agricultural domain.

nios.ac.in and various websites of the Department of Agriculture, Govt. of India, such as *farmer.gov.in*, *agricoop.nic.in*.

The terms extracted using the regular expressions are then weighted as per the following assumptions:

(i) A noun is preferred to other words which satisfy the same regular expressions.
(ii) High frequency words (except the stop words) are significant terms of the domain.
(iii) Words occurring with multiple patterns are given more weights in comparison with words occurring with single pattern only.

The set of candidate terms thus obtained also contained many irrelevant terms, which are then removed from the set with manual inspection. This process could not be automated, as no suitable threshold value could be obtained for segregating these terms from the relevant ones. Table 3 shows the scores of some valid and invalid terms in support of the above observation.

The vocabulary thus obtained is further expanded through extraction of composite terms (multiword term of length up to three words) from the input text. The length threshold for composite terms was chosen to be three based on inspection, and it was further corroborated by referring to AGROVOC.

Linguistic filters have been used to extract the composite terms. These filters are based on parts-of-speech (POS) combinations of words occurring in the text. The following linguistic filters have been used in this work:

- (NNP,NNP); (NNP,NNS); (NNP,NN); (NNS,NNS); (NN,NN); (NN,NN,NNS); (NN,NN,NN)- combinations of nouns.
- (JJ, NNP); (JJ, NN); (JJ, NN) – adjective followed by noun.

Composite terms in which at least one of the constituent words is contained in the list of candidate terms are included in the final list of terms. An overview of the algorithm is presented in Fig. 3.

3.2. Results

For the present work we executed the RENT algorithm on a test data of 200 pages of agricultural text taken from various Government websites as given in Section 3.1. The terms extracted are manually evaluated to calculate the precision of the algorithm using the following formula:

$$precision = \frac{\text{Number of valid terms extracted}}{\text{Total number of terms extracted}}$$

Precision of RENT algorithm for term extraction in the present work is 75.7%. Recall of the algorithm is calculated on random samples of data. Ten samples of 5 pages each have randomly been selected from the input text.

Recall has been calculated on these samples in 10 iterations in a cumulative way as explained below. The recall value at the kth ($1 \leq k \leq 10$) iteration is calculated as:

Table 4 gives the results for the 10 iterations. Average recall obtained on these samples is 65.27%.

$$recall_k = \frac{\text{No. of valid terms extracted by the algorithm from the first k sets of 5 pages}}{\text{No. of valid terms present in the those k sets}}$$

Table 4 – Recall of term extraction algorithm on random samples of data.

S. no.	Number of pages	No. of valid terms extracted by the algorithm	No. of valid terms present in the input text	Recall %
1	5	58	90	64.45
2	10	116	193	60.10
3	15	159	274	58.02
4	20	186	287	65.03
5	25	249	361	68.97
6	30	300	426	70.42
7	35	316	466	67.81
8	40	323	488	66.19
9	45	328	497	65.99
10	50	331	503	65.80

4. mOIE for relation extraction for ontology

Open Information Extraction [3–5] does not require any previous knowledge of the relations and works directly with the domain text. An OIE based scheme takes textual data as input, and produces the related terms along with their relations as its output. Hence, these methods involve extraction of both the pairs of related terms, and the relations existing between these pairs.

An OIE system identifies the relations between two terms by extracting and analyzing the text(s) occurring between these terms in the corpus. For the present work the OIE scheme is modified so as to get the relevant relations for ontology creation.

The scheme is implemented in two steps:

I. Identification of groups of related terms among a given set of terms.
II. Extraction of the textual patterns occurring between the related terms to discover the semantic relationships.

For Step I we use the terms extracted using the RENT along with domain text. Two approaches have been followed for relation extraction:

- Statistical approach based on word frequency distribution
- Semantics based approach using WordNet

The details of our experiments are given in the Sections 4.1 and 4.2. The WordNet based approach has been found to be the most effective in relation extraction. This scheme presented in Section 4.2 has been found to be useful for synonym extraction.

4.1. *Frequency distribution based relation extraction*

In the present work we have modified the dynamic programming based scheme given in [23] to find the pairs of related terms. The scheme defines two vectors for a given term w:

- *position vector* of w which is a vector $(p_1, p_2, \ldots, p_n)$ where n is the number of times the word w occurs in the document, and p_i is the position of its ith occurrence.
- *recency vector* of the word w which is computed from its position vector as the vector of length $n-1$, and is defined as $(p_2 - p_1, p_2 - p_3, \ldots, p_n - p_{n-1})$.

The intuitive idea has been that the related words will have similar recency vectors in a document. Following [23] we used three constraints:

4.1.1. *Starting point constraint*

This constraint uses the distance (difference between positions) between the first occurrences of two words. Words with a distance smaller than half of the length of the text under consideration satisfy this constraint.

$$|\textit{first occurence } (w_i) - \textit{first occurence } (w_j)| < \frac{1}{2} * (\textit{length of text})$$

For our experiments we have randomly chosen 20 agriculture terms and the relevant pages from the collection of input text mentioned in Section 3. Table 5 gives the terms and their starting points in the document.

The pairs which satisfy the starting point constraint are mostly related with each other in some way. Some examples of such pairs are (`barley`, `bajra`); (`cereal`, `maize`); (`brinjal`, `eggplant`); (`corn`, `maize`); (`seed`, `sowing`). These findings encourage us to proceed further with the second constraint, viz. the Euclidean distance constraint.

4.1.2. *Euclidean distance constraint*

In this constraint, the recency vectors v_1 and v_2, of two words w_1 and w_2 respectively are used to calculate the Euclidean distance between these two words, in terms of their means m_1, m_2, and standard deviations s_1, s_2. The words w_1 and w_2 are considered to be related if the distance is below certain threshold T, i.e.

$$\sqrt{(m_1 - m_2)^2 + (s_1 - s_2)^2} < T,$$

Although this constraint has been found to be very effective for word alignment between two parallel documents [23], however, when applied in agriculture domain for determining relatedness between terms it did not produce any useful result. In particular, when we applied this constraint on the set of terms satisfying the starting point constraint, no appropriate threshold value T could be found that can segregate similar terms. For illustration, as shown

Table 5 – Table showing 20 words along with their starting points.

w_i	Term	Starting point	w_i	Term	Starting point
w1	Bajra	1551	w11	forage	114
w2	Barley	1025	w12	plants	17
w3	Cereal	205	w13	irrigation	105
w4	Brinjal	4433	w14	lettuce	5694
w5	Cabbage	5693	w15	maize	249
w6	Corn	2119	w16	seed	443
w7	Crop	39	w17	soil	160
w8	Cultivation	74	w18	sowing	495
w9	Eggplant	5721	w19	weeds	450
w10	Fodder	7	w20	wheat	1442

Table 6 – Euclidean distance.

Pairs of terms	Distance
brinjal, eggplant	866.69
brinjal, irrigation	91.17
corn, maize	1702
corn, lettuce	903.35

in Table 6, the Euclidean distance between brinjal and eggplant is much more than the distance between brinjal and irrigation, although brinjal and eggplant are synonymous to each other. Similarly, despite being synonymous to each other corn and maize have a distance much higher than the distance between corn and lettuce. Hence, we did not use this constraint any further.

4.1.3. *Length constraint*

The length constraint suggests that two words w_1 and w_2 with $f1$ and $f2$ as the respective frequency of occurrence in the input text are similar if $\frac{1}{2} * f_2 < f_1 < 2 * f_2$.

We experimented with all the terms qualifying the starting point constraint. However, the constraint has not been found to be much effective in identifying similarity between terms. For instance, the pair (eggplant, lettuce) satisfies the length constraint, whereas the pair (brinjal, eggplant) does not satisfy this constraint despite being synonymous to each other.

Applications of these three constraints on the sample words and text clearly established that these statistics-based heuristics are not useful for relationship identification from a given text. However, we have used position vectors for extraction of is_intercrop relations as explained in Section 5.1.4.

4.2. *WordNet based semantic similarity*

In WordNet [24] lexical concepts are represented by synonym sets, known as *synsets*, which share a common meaning. A synset consists of English noun, verbs, adjectives, and adverbs.

Typically, a synset is identified by a 3-part name of the form: word.pos.nn,[6] where,

- word is the specific word to which the synset belongs,
- pos is the part of speech of the synset, and
- nn is the number associated with the specific synset.

For illustration, synsets for agriculture are:

- agribusiness.n.01,
- farming.n.01,
- department_of_agriculture.n.01,
- agriculture.n.04.

In these experiments, we have used the WordNet path similarity measure [25] to group similar terms. The shorter is the path between two synsets, the higher is their similarity value. Typically, path similarity values lie between 0 and 1, where 1 means absolute similarity, and 0 means no similarity. In the present work, similarity between two words w_1 and w_2 is obtained as follows:

First of all the synsets of w_1 and w_2 are obtained. Assuming the number of synsets that w_1 and w_2 have to be n and m, respectively, we denote the synsets of w_1 as $\{S_{1i} \mid i = 1, 2, \ldots, n\}$ and the synsets of w_2 as $\{S_{2j} \mid j = 1, 2, \ldots, m\}$. We obtain the similarity between w_1 and w_2 in two steps:

(i) In this step the path similarity between each pair of synset (SS_{1i} and SS_{2j}), for $i = 1, 2, \ldots, n$, and $j = 1, 2, \ldots, m$ is calculated. Let it be denoted by $path\_sim(SS_{1i}, SS_{2j})$.

(ii) Semantic similarity between w_1 and w_2 is calculated as:

$$sim(w_1, w_2) = \max_{i,j}(path\_sim(SS_{1i}, SS_{2j}),\ i = 1, 2, \ldots, n;\ j = 1, 2, \ldots, m)$$

Analysis of WordNet for grouping related terms is done using 200 terms extracted from the crops sub-domain using the RENT algorithm. Out of the 200 terms extracted 70 are composite terms and therefore could not be processed using WordNet as synsets for composite terms are not present in the WordNet. Further, there are 28 single word terms, e.g. foodgrain, berseem, intercrop, nigerseed, masur,[7] for which no synsets are present in the WordNet. Remaining 102 terms have therefore been subjected to the path-

[6] http://www.nltk.org/howto/wordnet.html.

[7] Last searched on 18/11/2017.

Table 7 – Related terms and text occurring between them.

Term 1	Text occurring between term 1 and term 2	Term 2	Similarity value
Eggplant	also called as	aubergine	1
Eggplant	or	brinjal	1
Forage	crops and biodiversity include legumes like desmodium, lablab, stylosanthes, vigna, macroptelium, centrosema, etc.; trees, bushes and ,range	grass	1
Grass	these terms do not occur in a common sentence	weeds	1
Groundnut	also known as	peanut	1
Cabbage	these terms do not occur in a common sentence	lettuce	1
Agriculture	these terms do not occur in a common sentence	cultivation	0.5
Cereal	viz. maize, Like	barley	0.5
Legume	, viz. cowpea and cluster- particularly sylosanthes, siratro, lablab	bean	0.5
Seed	of lablab	bean	0.5
Cassava	these terms do not occur in a common sentence	starch	0.5
Cereal	And	grass	0.5
Cereal	(rice,	wheat	0.5

similarity based measure for discovering relatedness. It is observed that the words having similarity values $\geq$0.5 are closely related.

Step II of mOIE focuses on extracting the text between related pairs obtained in step I. These text(s) are then analyzed to discover the patterns for identification of relations between the corresponding terms. Table 7 shows some words having similarity value greater than or equal to 0.5. It brings on following important observations:

(i) For synonyms the semantic similarity value is equal to 1.
(ii) Semantic similarity equal to 1 does not necessarily mean the words are synonyms.

Term pairs which are closely related also have semantic similarity equal to 1. These terms occur in proximity with each other but they do not contain any specific textual pattern between them. E.g. (`forage`, `grass`) are closely related but these are not synonyms.

(iii) WordNet Similarity value between 0.5 and 1.0 identifies related terms but they do not contain any specific text between them. Moreover, these pairs are linked by different types of relations. For example,
- `cultivation` is related to `agriculture` but it is just a process carried out in agricultural sector [26],
- `barley` is a `cereal` [27],
- `bean` is a type of `legume` [28].

The results suggest that mOIE scheme is effective in identifying synonyms using WordNet similarity measure and a constraint that the term pairs should occur separated by a positional distance in the range of [2,10].

4.3. *Results*

In our experiments mOIE performed well for identification of synonym relation with a precision of 67% and recall of 72% on 200 pages of agricultural data. It is also concluded that mOIE is not suited for extraction of other relations, whether domain or semantic. Moreover, although the mOIE scheme proved useful for identification of related terms but the specific relations holding between these terms could not be inferred from the text occurring between these terms. Few such examples are shown in Table 7. This leads us to extend the scheme using domain-specific knowledge, wherein the specific relations which are to be extracted from the domain text are pre-decided. The algorithm is responsible for identifying the term pairs satisfying these relations. The proposed scheme has been named RelExOnt, for identification of four relations: `has_synonym`, `is_type_of`, `is_a`, and `is_intercrop`. Details of these relations are explained in Section 5.1. RelExOnt uses the observations made with our experiments with WordNet based similarity, explained in Section 4.2, for extraction of synonyms. In addition, expert knowledge is used for framing the constraints for identification of related terms for each of

the four relations. Section 5 elaborates the RelExOnt algorithm in details.

5. RelExOnt: Relationship extraction for ontology

5.1. The proposed scheme

In this technique the possible set of relations holding between the domain terms are selected using expert knowledge and the corresponding terms are identified. Often it involves specification of some rules/constraints for identification of a certain relation holding between two terms [13]. The terms related by the already identified relations are extracted using the specified constraints for each relation. The following set of relations has been considered for the terms belonging to the crops sub-domain:

- `is_a`: identifies the concept-instance pairs
- `is_type_of`: specifies a hierarchical relationship
- `has_synonym`: describes an equivalence relationship
- `is_intercrop`: Intercrop [27] is defined as a crop which when grown with plants of different kinds increases the yield.

Table 8 shows some example term-pairs satisfying above relations.

Since the present work deals with four relations viz. `has_synonym`, `is_type_of`, `is_a` and `is_intercrop`, the value of r is 4, and the list rel consists of four elements: ('`has_synonym`', '`is_type_of`', '`is_a`', '`is_intercrop`'). Sections 5.1.1-5.1.4 discuss identification of the pair of terms satisfying these relations in detail.

5.1.1. Identification of equivalent terms: has_synonym relation

This relation identifies the synonymous terms. This relation satisfies two axioms:

- Transitivity – i.e. for any three terms x, y, z *has_synonym*(x, y) $\bigwedge$ *has_synonym*(y, z) $\rightarrow$ *has_synonym*(x, z)
- Symmetricity – i.e. for any two terms x, y *has_synonym*(x, y) $\rightarrow$ *has_synonym*(y, x)

Two constraints are framed for identification of synonymous terms using the observations inferred in Section 4.2. Hence the value of m_1 is 2. These constraints are given below: For its application we take inputs from the results obtained in Section 5.2 on WordNet based semantic similarity. As even non-synonymous terms may have WordNet semantic similarity score equal to 1, further filtration of the term pairs is required. We have used position vector based heuristic for the said purpose. It works as follows:

Constraint 1:	*sim*(*terms*[j], *terms*[k]) = 1, where *sim*(*terms*[j],*terms*[k]) = WordNet Similarity of two different terms, as explained in Section 4.2.
Constraint 2:	$2 \leq d(terms[j], terms[k]) \leq 10$, where *d* is the positional distance between terms[j] and terms[k]. This constraint filters out the invalid candidate synonyms obtained from Constraint 1.

The overall framework for RelExOnt algorithm is given in Fig. 4. The algorithm works as follows:

- The names and number of relations to be extracted is identified, names of relations are stored in the list named `rel` and number of relations is stored in the variable r.
- For each relation in `rel`, constraints (rules) for identifying the relation, `rel[i]`, are specified, two terms are identified as related by `rel[i]` when they satisfy all the constraints specified for `rel[i]`.

First the distance between two terms, t_1 and t_2, having position vectors v_1 and v_2, respectively, is computed as follows:

$$d(t_1, t_2) = \min(|v_1(i) - v_2(j)|)$$

where $i = 1 \ldots l_1$, $j = 1 \ldots l_2$, $l_1 = length\ (v_1)$, $l_2 = length\ (v_2)$. The pair of terms (t_1, t_2) is considered to be synonyms if $d(t_1,t_2)$ lies in [2,10].

For illustration consider Table 9 showing position vectors for some sets of terms whose WordNet similarity value is 1.

Table 10 shows the value of $d(t_1,t_2)$ for five pairs of terms. It can be easily inferred that (`brinjal`, `eggplant`) and (`maize`,

Table 8 – Sample relationships identified.

X (first term)	Relationship	Y (second term)
Potato	is_a	Tuber Crop
Rabi crop	is_type_of	Crop
Kharif crop	is_type_of	Crop
Brinjal	has_synonym	Aubergine
Sugarcane	is_intercrop	Potato

```
i. Choose the semantic relationships that are to be extracted from text
into a list named rel. Let the number of chosen relationships be 'r'.

ii. identify 'a' axioms holding on 'r' relations

iii. for i=1 to r:
      identify m_i constraints holding for rel[i]
      for j=1 to n-1:
           for k =j+1 to n:
                if terms[j] and terms[k] satisfy all the m_i constraints
                for rel[i]
                     then append the pair of terms[j] and terms[k] to
                     rel_terms_i.
                else:
                     continue.
iv. apply 'a' axioms to rel_terms_i

Here, rel=the list containing the chosen semantic relations
      m_i=number of constraints that hold for i^th relation.
      terms[]=list of terms.
      n=number of terms in terms[]
      rel_terms_i=list containing all those pairs of terms which are
      related by i^th relation.
```

Fig. 4 – The RelExOnt algorithm.

Table 9 – Synonyms set obtained using Constraint 1 with their position vectors.

Synonyms set	Terms	Position vector
1	brinjal	4433, 10325, 10330, 10344, 10391
	aubergine	10387
	eggplant	5721, 10332, 10360
2	seed	443, 560, 567, 594, 1712, 3430, 3439, 5709, 7062, 7103, 8170, 8560, 8612
	sowing	495, 867, 1454, 2978, 6562, 11810
3	maize	249, 677, 912, 1101, 1140, 1240, 2621, 2663, 2664, 2696, 2883, 3919, 4232, 6818, 6942, 6954, 7462, 7490, 7494, 7516, 7519, 7554, 7629, 7646, 8153, 8405, 8518, 8545, 8553, 8595, 8634, 8647, 8801, 8831, 10071, 11221
	corn	2119, 2624, 5725
4	weed	450, 10132, 11709
	grass	214
5	cabbage	5693, 5937
	Lettuce	5694, 5940, 10292

Table 10 – $d(t_1, t_2)$ values for four pairs of terms.

t_1	t_2	$d(t_1,t_2)$
brinjal	eggplant	2
Maize	Corn	3
Cabbage	lettuce	1
Seed	sowing	52
Weed	Grass	236

corn) are synonym pairs, while (seed, sowing), (cabbage, lettuce) and (weed, grass) are not.

5.1.2. *Identification of hierarchical relation: is_type_of*

This is a hierarchical relation, where we identify the terms as type of other terms in the ontology. For example, cereal fodder is a type of fodder. This relation follows the subclass-superclass structure, as the properties of superclass are inherited by its subclass. E.g., cereal fodder inherits all the

Table 11 – Examples for which is_type_of([t1 t2], t2) holds.

[t1 t2]	t2
[rabi crop]	crop
[food grain]	grain
[coarse grain]	grain
[coarse cereal]	cereal
[cereal fodder]	fodder
[drip irrigation]	irrigation
[sustainable agriculture]	agriculture

properties of fodder. It can be clearly observed that this relationship is an asymmetric one, i.e. if x is of type y then y is surely not of type x.

For is_type_of relation, m2 = 1, i.e. there is only one constraint:

- ∀ composite terms [t_1 t_2], if $t_2 \in$ {'crop', 'fodder', 'fertilizer', 'agriculture', 'irrigation', 'cereal', 'grain', 'soil'}, then the relation *is_type_of* ([t_1 t_2], t_2) holds.

Table 11 shows some examples for which this relation holds.

However, we found a few examples for which *is_type_of* ([t_1 t_2], t_2) does not hold are:

- (sweet potato, potato),
- (sesbania sesban, sesban),
- (panicum maximum, maximum).

5.1.3. *Identification of instances: is_a relation*

This relation is used to identify instances of a particular concept in an ontology. For illustration is_a(x, y)⇒x is an instance of y, e.g., is_a(tomato, vegetable), is_a(potato, tuber_crop). It can be easily verified that:

- This relation is asymmetric.
- It is transitive, i.e. $\forall x, y, z \in terms$, $is\_a(x, y) \wedge is\_a(y, z)$ $is\_a(x, z)$.

For illustration, is_a(mustard, oilseed) and is_a(oilseed, crop) implies that mustard is a crop, i.e. is_a(mustard, crop) holds.

One constraint holds for is_a relation, hence m3 = 1.This constraint uses two patterns to identify the is_a relation. Table 12 shows the two patterns.

While applying the patterns to identify is_a relation from the text, following points need to be taken care of in selecting appropriate 'x' and 'y's taking into consideration one practical aspect, viz. if the candidate term x is a composite one, then the string of maximum possible length should be considered.

5.1.4. *Identification of intercrops: is_intercrop relation*

The relation is_intercrop(x, y) means x is an intercrop [29] with y. For example, is_intercrop (soybean, cotton) means cotton can be grown as an intercrop with soybean to have better yield for both. Clearly, it is a symmetric relation, i.e.

$\forall (x, y) \in terms$, $is\_intercrop(x, y) \rightarrow is\_intercrop(y, x)$

To identify this relation, following constraints have to be fulfilled.

Constraint 1: The objective of this constraint is to find out all those terms which are occurring in the neighbourhood of the word intercrop or one of its morphological variations, i.e. intercropped, intercropping, intercrops. For this work we have used the distance threshold to be 10 to define the neighbourhood. However, all the agricultural terms occurring within the neighbourhood need not be intercrops. To fulfill the is_intercrop relationship the primary condition is that both of the two terms have to be crops. For example, a fertilizer cannot be in is_intercrop relation with any agriculture term. Constraint 2, given below, takes care of the above.

Constraint 2: $\forall (x, y) \in terms$ *is_intercrop*(x, y) *iff* *is_a*(x, *crop*) *and* *is_a*(y, *crop*)

For illustration, the sample text file taken from [30] contains 4 occurrences of intercrop, including different morphological variations. We store the word positions in a variable *v_pos*. For the sample file we have *v_pos* = [887, 36,070, 60,090, 60,112]. Table 13 provides the list of terms occur within a distance of 10 from these four occurrences.

Table 12 – Patterns and example text for is_a relation.

Pattern	Example text	Relation
$t_a(t_{b1}, t_{b2}, t_{b3}, \ldots t_{bn})$	foodgrains (rice, wheat, maize, millet, pulses)	is_a(rice, foodgrain) is_a(wheat, foodgrain) is_a(maize, foodgrain) is_a(millet, foodgrain) is_a(pulse, foodgrain)
t_a*like* $t_{b1}, t_{b2}, t_{b3}, \ldots t_{bn}$	fodder crops like berseem, lucerne, turnip, etc.	is_a(berseem, foddercrop) is_a(lucerne, foddercrop) is_a(turnip, foddercrop)

Table 13 – Example text [30] for is_intercrop relation.

v_pos[i]	text at *v_pos*[i]±10
887	April 2010. Fig.4. 8: BHOOCHETNA Maize Intercropping with Red Gram Karnataka Agricultural Production and Programmes99Table 4
60090	areca nut. The elephant foot yam is widely grown as intercrops in litchi, coconut, banana, orchards. Spices like black pepper, ginger
60112	vanilla, nutmeg, clove and some medicinal plants are the ideal intercrops for coconut . Agricultural Research, Education and Extension161Hi-tech horticulture and

Table 14 – Results of RelExOnt on 10 Random Samples of Data.

Sample No.	1	2	3	4	5	6	7	8	9	10
Precision Value (%)	83.34	83.34	87.50	92.31	86.36	90.00	95.34	90.00	94.12	66.67
Avg. precision value (%)	86.89									

RelExOnt extracts the following pairs of terms satisfying `is_intercrop` relation from the text given in Table 13.

- `is_intercrop (red gram, maize)`,
- `is_intercrop (elephant foot yam, litchi)`,
- `is_intercrop (elephant foot yam, coconut)`,
- `is_intercrop (elephant foot yam, banana)`,
- `is_intercrop (elephant foot yam, orchards)`,
- `is_intercrop (elephant foot yam, black pepper)`,
- `is_intercrop (elephant foot yam, ginger)`,
- `is_intercrop (vanilla, coconut)`,
- `is_intercrop (nutmeg, coconut)`,
- `is_intercrop (clove, coconut)`.

5.2. *Results*

Performance of the relationship extraction algorithm, RelExOnt, is measured in terms of precision calculated with respect to the relations extracted. Expert opinion is used to judge whether the relation extracted between two particular terms does hold in real life. Precision for relation extraction are calculated as follows:

$$precision = \frac{number\ of\ correctrelated\ pairs\ of\ terms\ extracted}{total\ number\ of\ related\ pairs\ of\ terms\ extracted}$$

We have evaluated the performance of the scheme on 10 random samples, each consisiting of 20 pages of agricultural text. Table 14 shows the precision values.

We have used Protégé,[8] an ontology editing software for generation of owl file for the ontology. The extracted terms and relations are organized into .csv files in order to feed to protégé for generating .owl file for the resultant ontology. In a similar way, Protégé can be used to generate .rdf file for the ontology. Graphical view of a part of the ontology generated using the terms and relations extracted using the proposed scheme is shown in Fig. 5.

[8] http://protege.stanford.edu/.

6. Conclusion

Development of ontology is an important aspect of modern day information processing. Although various tools, e.g. Protégé and owlready package in python, exist for editing and managing ontologies, no tool is available for automatic creation of ontology from domain text. The present paper aims at developing techniques for automatic extraction of vocabulary and relationships between the terms which is fundamental for ontology creation.

In this paper we established a baseline algorithm mOIE which works on WordNet-based similarity to identify different relationships. However, our experiments suggest that this scheme is effective in identifying the `has_synonym` relations from a given input text. Identification of other relations require more domain knowledge to be imparted to the system. The mOIE scheme is further modified to extract three more relations namely, `type_of`, `is_a`, `is_intercrop`. The reason behind choosing these relations is that these have been found to be important from practical as well as ontology point of view. The new knowledge-based scheme is named as RelExOnt.

The algorithm does not use any, dictionary or thesaurus type domain-specific knowledge source for identification of terms and relations. As a consequence, the set of terms extracted using this algorithm is not exhaustive for crops sub-domain. However, as newer documents are available, more and more terms can be accumulated along with their relationships with other domain related terms and put them in the ontology through matching and merging. Our present work is directed in this direction.

The research work carried out in this paper has two limitations:

- Number and type of relations identified is restricted by the input text. For instance, `grows_in_soil`, `grows_in_weather` could not be identified during this work because of lack of such relations in the input text.
- Another challenging aspect of relation extraction in agriculture domain is evaluation of recall for the proposed

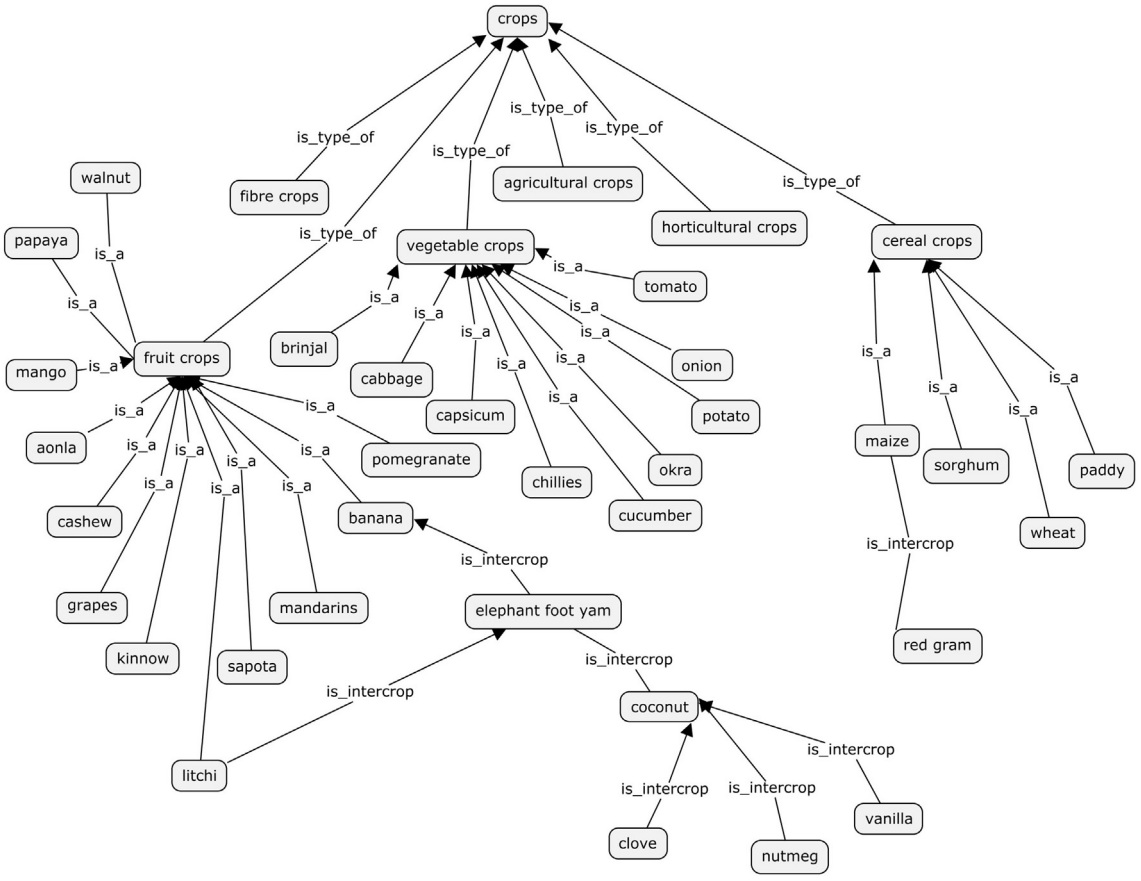

Fig. 5 – Partial view of the resulting ontology.

scheme. Recall calculation for relation extraction with respect to gold standard involves identification of all the terms-relation pairs from the input text. This is not a feasible solution with limited resources and potentially infinite volume of available texts. We therefore initially resorted to recall calculation using terms present in AGROVOC. However, this has to be abandoned as majority of the terms present in AGROVOC are different from the terms present in the texts collected from the government and other documents collected for this work.

In future, we plan to expand this algorithm by incorporating a larger vocabulary set from agriculture domain. We also aim at developing an incremental algorithm to merge smaller ontologies into a bigger one in order that it can cater to domain-specific and inter sub-domain query processing systems for agriculture domain.

REFERENCES

[1] Gruber TR. Toward principles for the design of ontologies used for knowledge sharing. Int J Hum Comput Stud 1995;43 (5):907–28.
[2] Chatterjee N, Kaushik N. RENT: regular expression and NLP-based term extraction scheme for agricultural domain. In: Proceedings of the international conference on data engineering and communication technology. Springer; 2017. p. 511-11.
[3] Banko M, Cafarella MJ, Soderland S, Broadhead M, Etzioni O. Open information extraction from the web. In: International joint conference on artifical intelligence, Hyderabad; 2007. p. 2670–76.
[4] Fader A, Soderland S, Etzioni O. Identifying relations for open information extraction. In: Proceedings of the conference on empirical methods in natural language processing, association for computational linguistics; 2011. p. 1535–45.
[5] Mausam, Schmitz M, Bart R, Soderland S, Etzioni O. Open language learning for information extraction. In: Proceedings of the 2012 joint conference on empirical methods in natural language processing and computational natural language learning, association for computaional linguitics, Korea; 2012. p. 523–53.
[6] Byrne K. Relation extraction for ontology construction. Unpublished doctoral dissertation. Univ. of Edinburgh; 2006.
[7] Maynard D, Funk A, Peters W. SPRAT: a tool for automatic semantic pattern-based ontology population. In: Proceedings of the international conference for digital libraries and the semantic web; 2009. p. 1–15.
[8] Brinjal Farming Information in India. <http://www.agrifarming.in/brinjal-farming/>; 2017.
[9] Al-Rajebah NI, Al-Khalifa HS. Semantic relationship extraction and ontology building using wikipedia: a comprehensive survey. Int J Comput Appl 2010;12(3):6–12.
[10] Assis PH, Casanova MA. Distant supervision for relation extraction using ontology class hierarchy-based features. In:

Presutti V, Blomqvist E, Troncy R, Sack H, Papadakis I, Tordai A, editors. The semantic web: ESWC 2014 satellite events. Springer International Publishing. p. 467–71.
[11] Alfonseca E, Filippova K, Delort JY, Garrido G. Pattern learning for relation extraction with a hierarchical topic model. In: Proceedings of the 50th annual meeting of the association for computational linguistics; 2012. p. 54–5.
[12] Lang J, Lapata M. Similarity-driven semantic role induction via graph partitioning. J Comput Linguist 2014;40(3):633–69.
[13] Konstantinova N. Review of relation extraction methods: what is new out there? In: International conference on analysis of images, social networks and texts. Springer International Publishing; 2014. p. 15–28.
[14] Hearst MA. Automatic acquisition of hyponyms from large text corpora. In: Proceedings of the 14th conference on computational linguistics, association for computational linguistics, Morristown; 1992. p. 539–45.
[15] Bastista DS, Martins B, Silva MJ. Semi supervised bootstrapping of relationship extractors with distributional semantics. In: Proceedings of the 2015 conference on empirical methods in natural language processing, association of computational linguistics, Portugal; 2015. p. 499–504.
[16] Zhang X, Gao Z, Zhu M. Kernel methods and its application in relation extraction. In: Proceedings of the international conference on computer science and service system, IEEE; 2011. p. 1362–5.
[17] Kambhatla N. Combining lexical, syntactic, and semantic features with maximum entropy models for extracting relations. In: Proceedings of the ACL 2004 on interactive poster and demonstration sessions, association for computational linguistics, Morristown; 2004. p. 22.
[18] Miller S, Fox H, Ramshaw L, Weischedel R. A novel use of statistical parsing to extract information from text. In: Proceedings of the 1st North American chapter of the association for computational linguistics conference, Washington; 2000. p. 226–33.
[19] Culotta A, McCallum A, Betz J. Integrating probabilistic extraction models and data mining to discover relations and patterns in text. In: Proceedings of the main conference on human language technology conference of the North American chapter of the association of computational linguistics, New York; 2006. p. 296–303.
[20] Riedel S, Yao L, McCallum A, Marlin BM. Relation extraction with matrix factorization and universal schemas; 2013. <http://works.bepress.com/benjamin_marlin/1/>.
[21] Mintz M, Bills S, Snow R, Jurafsky D. Distant supervision for relation extraction without labeled data. In: Proceedings of the joint conference of the 47th annual meeting of the ACL and the 4th international joint conference in natural language processing of the AFNLP, association for computational linguistics; 2009. p. 1003–11.
[22] Sorokin Daniil, Gurevych. Context aware representations for knowledge base relation extraction. In: Proceedings of the 2017 conference on empirical methods in natural language processing; 2017. p. 1785–90.
[23] Chatterjee N, Agrawal S. Word alignment in English-Hindi parallel corpus using recency-vector approach: some studies. In: Proceedings of the 21st international conference on computational linguistics and the 44th annual meeting of the association for computational linguistics, USA; 2006. p. 649–8.
[24] Miller GA. WordNet: a lexical database for English. Commun ACM 1995;38(11):39–41.
[25] http://www.nltk.org/howto/wordnet.html.
[26] https://en.wikipedia.org/wiki/Cultivation.
[27] https://www.britannica.com/plant/barley-cereal.
[28] http://www.erinnudi.com/2014/10/22/difference-beans-legumes/.
[29] Mousavi SR, Eskandari H. A general overview on intercropping and its advantages in sustainable agriculture. J Appl Environ Biol Sci 2011;1(11):482–6.
[30] State of Indian Agriculture; 2011-12, p. 160. <http://agricoop.nic.in/SIA111213312.pdf>.