

# SEARCH FOR ENTITIES BASED ON THE IMPLICIT SEMANTIC RELATIONS

TRAN LAM QUAN<sup>1,\*</sup>, VU TAT THANG<sup>2</sup>

<sup>1</sup>*Research and Implementation Center, Vietnam Airlines*

<sup>2</sup>*Institute of Information Technology, Vietnam Academy of Science and Technology*

\*[quantl@vietnamairlines.com](mailto:quantl@vietnamairlines.com)



**Abstract.** Search for the entity based on implicit semantic relations is to find out information/knowledge from an unfamiliar semantic domain using similarities from the familiar one through query. The paper presents extracting, clustering, ranking techniques and a model of search for entity based on implicit semantic relation on Vietnamese language domain.

**Keywords.** Implicit Relational Entity Search; Named-Entity; Semantic Relation; Similarity Relation.

## 1. INTRODUCTION

In fact, there are always relations between two entities existing such as: Khuê Văn các – Văn miếu (Temple of Literature); Stephen Hawking – Physicist; Thích Ca – School of Mahāyāna (phái Đại thừa); Apple – iPhone; etc.

Moreover, there are similarity relations between two pairs of entities, for example, Nguyễn Du – Truyện Kiều (The Tale of Kieu) and Đoàn Thị Điểm – Chinh phụ ngâm (Lament of the soldier's wife), these pairs of entities are semantically similar: it is “the author”; The similarity of the semantic relation between Hanoi – Vietnam and Paris – France is “the capital”; For The Qurán – Islam and The Gospel – Christianity, it is “the Bible”; For the Cốm – làng Vòng và Chả mực – Hạ Long (green rice - Vong village and Grilled chopped squid – Halong), it is “the specialty”; etc. Each semantic relation is hidden under a particular meaning.

In real life, there are questions like: “If Fansipan is the highest mountain in Vietnam, which one is the highest in India?” or: “If Trump is the current president of the United States, who is the most powerful in Swedish?”, etc.

By the keyword search engine, according to statistics, queries are often short, ambiguous, and poly-semantic [5]. Approximately 71% of web search queries contain names of entities, as statistics [3]. If the user enters the entities like “Vietnam”, “Hanoi”, “French”, then the search engine only results in documents that contain the above keywords, but does not immediately answer “Paris”. Because of looking for entities only, query extending and query rewriting techniques are not applied to the type of the implicit semantic relation in the entity pair. Therefore, a new search morphology is researched. The pattern of the search query is in the form of:  $(A, B), (C, ?)$ , in which  $(A, B)$  is the source entity pair,  $(C, ?)$  is the target entity pair. At the same time, the two pairs  $(A, B), (C, ?)$  have a semantic similarity. In other words, when the user enters the query  $(A, B), (C, ?)$ , the search engine has the duty

of listing entities  $D$  so that each entity  $D$  satisfies the condition of semantic relation with  $C$  as well as the pair  $(C, D)$  have similarity relation with the pair  $(A, B)$ . Semantic relation - in the narrow sense and in the lexical perspective - is expressed by terms/patterns/context surrounding (front, middle and behind) the known entity pair [4].

The pattern search morphology is called the Implicit Relational Entity Search or Implicit Relational Search (IRS). The ability to infer undefined information/knowledge by similar inference is one of the natural abilities of human. The paper aims to study and simulate the above ability. The IRS model searches for undefined information/knowledge from an unfamiliar domain using similarities from familiar domains. Because the semantic relation and similarity relation are not explicitly stated in the query (the query consists of only three entities:  $A, B, C$ ), the IRS model is called an implicit semantic entity search model. The contributions of the paper include the researches and implementation of the IRS system on the Vietnamese language domain, applied to 9 classes of entities (described in Section 4), giving a similarity measure in combination between terms and distributional hypothesis; thus, applied to the clustering algorithm to improve cluster quality.

The paper consists of four main sections. Section 1 is introduction. Section 2 presents the related works. The implicit relational entity search model is described in Section 3. Section 4 shows the data, experimental results, conclusions and directions.

## 2. RELATED WORK

Identifying the similarity relation between the pair of entities  $(A, B)$ ,  $(C, ?)$  is a necessary condition for determining the entity to be sought. As a problem of NLP (Natural Language Processing), relational similarity is one of the most important tasks of search for entities based on the implicit semantic relations. Thus, the paper lists the main research directions for relational similarity.

### 2.1. Structure mapping theory (SMT)

SMT [2] considers the similarity as a mapping of knowledge from the source domain to the target domain, according to the mapping rules: Eliminate the attributes of the object but maintain the relational mapping between objects from the source domain to the target domain.

- Mapping rules  $M : s_i \rightarrow t_i$ ; (in which  $s$ : source;  $t$ : target).
- Eliminate attribute  $\text{HOT}(s_i) \rightarrow \text{HOT}(t_i)$ ;  $\text{MASSIVE}(s_i) \rightarrow \text{MASSIVE}(t_i), \dots$
- Maintain relational mapping  $\text{Revolves}(\text{Planet}, \text{Sun}) \rightarrow \text{Revolves}(\text{Electron}, \text{Nucleus})$ .

Figure 1 shows that due to the same S (subject), O (object) structures, the SMT considers the pairs (Planet, Sun) and (Electron, Nucleus) are relation similarity, regardless of the fact that the source and target pairs - Sun and Nucleus, Planet and Electron are very different in properties (HOT, MASSIVE, ...).

Referring to the purpose of the paper, if the query is ((Planet, Sun), (Electron, ?)), SMT will output the correct answer: "Nucleus". However, SMT is not feasible with low-level structures (lack of relations). Therefore, SMT is not feasible with the problem of searching entities based on implicit semantic relation.

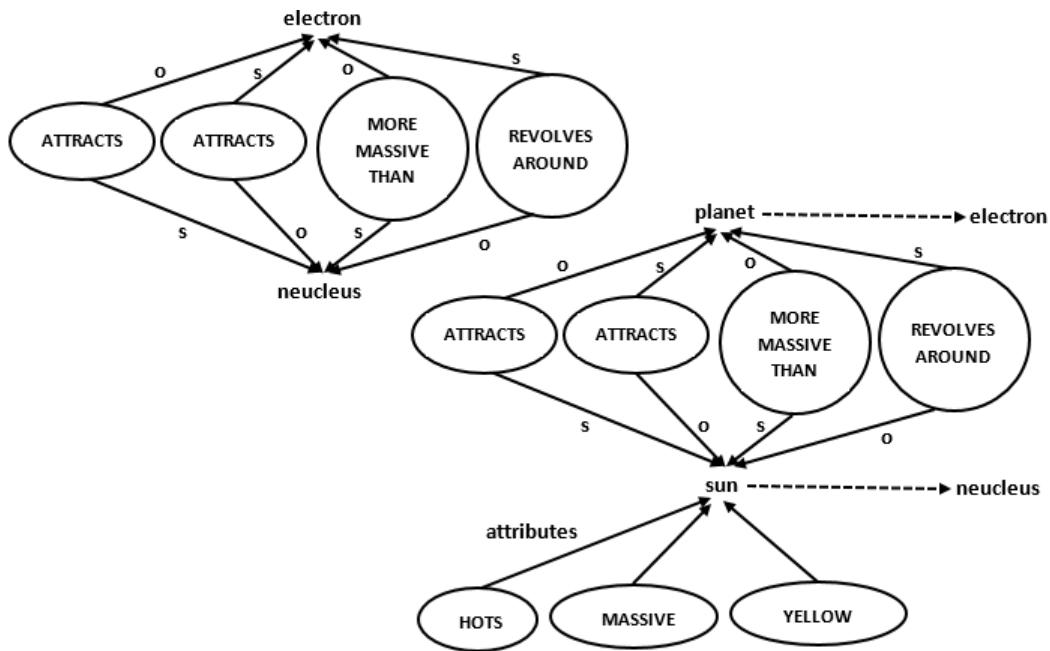


Figure 1. Structure mapping theory (SMT)

## 2.2. Vector space model (VSM)

Using the vector space model, Turney [11] presents the concept of each vector formed by a pattern containing the entity pair  $(A, B)$  and the occurrence frequency of the pattern. The VSM performs the relational similarity measurement as follows: Patterns are generated manually and queried to the Search Engine (SE), the number of results returned from the SE is the frequency of occurrence of such patterns. Thus, the relational similarity of two pairs of entities is computed by Cosine between two frequency vectors. For example, considering the pair (traffic, street) and the pair (water, riverbed), these two pairs are likely to appear in sentences, such as “traffic in the street” and “water in the riverbed”. Cosine measure between the two vectors (traffic, street) and (water, riverbed) will determine whether the two vectors are similar or not.

## 2.3. Latent relational analysis (LRA)

By extension of VSM, Turney combines it with LRA to determine level of relational similarity [12]. Like VSM, LRA uses a vector made up of the pattern/context containing the entity pair  $(A, B)$  and the frequency of the pattern (pattern in  $n$ -grams format). At the same time, LRA applies a thesaurus to extend the variants of:  $A$  bought  $B$ ,  $A$  acquired  $B$ ;  $X$  headquarters in  $Y$ ,  $X$  offices in  $Y$ , etc. LRA applies the most frequent  $n$ -grams to assign the pattern with the entity pair  $(A, B)$ , then builds a pattern – entity pair matrix, where each element of the matrix represents the frequency of the pair  $(A, B)$  in the pattern. In order to reduce the matrix dimension, the LRA uses Singular Value Decomposition (SVD) to reduce the number of columns in the matrix. Finally, the LRA applies a Cosine measure to define the relational similarity between two pairs of entities.

In spite of an effective approach to identifying relational similarity, LRA requires a long time to compute and process. LRA requires 8 days to perform 374 SAT analogy questions [1]. This is impossible with a real-time response system.

#### 2.4. Latent semantic relation

Bollega, Duc. et al. [1, 10], Kato [8] use the distributional hypothesis at the context level: In the corpus, if two contexts  $p_i$  and  $p_j$  are different but usually co-occur with entity pairs  $w_m, w_n$ , they are similar in semantics. When  $p_i, p_j$  are semantically similar, entity-pairs  $w_m, w_n$  are similar in relation.

The distribution hypothesis requires pairs of entities to always co-occur with contexts, and the Bollega clustering algorithm is proposed at the context level rather than clustering at the term level in the sentence. Measure of similarity based on the distribution hypothesis, which is not based on term similarity, will significantly affect the quality of the clustering technique, thus affecting the quality of the search system. The paper deals with the viewpoint of latent semantics, but adds Cosine measure at term level to improve clustering algorithm. The study is expanded by implementing the implicit semantic entity search model on the Vietnamese language domain.

In addition, the authors [1, 8, 10], do not consider the number of relations of the source and target pairs as uncertain in the sense of relational mapping. For example, we have a one-to-one relation when considering the pair of entities (Moon, Earth). Looking at the entity pair (Sun, Satellite), we have a one-to-many relation. Considering the entity pair (Manufacturer-Company) we have a many-to-many relation. If we apply three types of relational mapping to the entity search with the elements of time, the search results will be more accurate and up-to-date.

#### 2.5. Relational similarity based on Wordnet classification system

Veale [13] and Cao [6] proposed relational similarity measure based on similarity classification system in Wordnet. However, as mentioned above, Wordnet does not contain named entities. Thus, Wordnet is not suitable for entity search model.

From the existing issues, the paper researches the model of search for entities based on implicit semantic relations.

### 3. SEARCH FOR ENTITIES BASED ON IMPLICIT SEMANTIC RELATIONS

#### 3.1. Definition – general structure

The concept of searching entities through implicit semantic relation is the most obvious distinction for search engines based on keywords. Figure 2 simulates a query consisting of only three entities, query = (Vietnam, Mekong), (China, ?). Write the convention:  $q = (A, B), (C, ?)$ , where (Vietnam, Mekong) is a pair of source entities, (China, ?) is a pair of target entities. The search engine is responsible for identifying the entity (“?”) that has a semantic relation with the “China” entity, and the entity pair (China, ?) must be similarly related to the entity pair (Vietnam, Mekong). Note that the above query does not explicitly contain the semantic relation between the two entities. This is because semantic relations

are expressed in various ways around the pair of entities (Vietnam, Mekong), such as “the longest river”, “big river system”, “the largest basin”, etc. Due to the fact that the query consists of only three entities that do not include semantic relations, the model is called the implicit semantic relation search model.

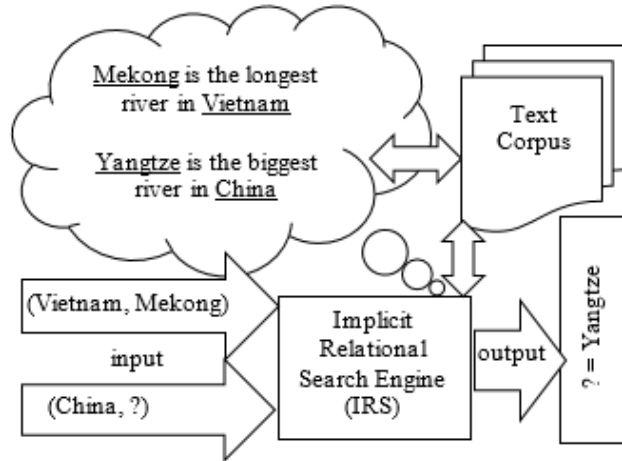


Figure 2. Implicit semantic relation search with input consisting of 3 entities

The morphology of search for entities based on implicit semantic relations must determine the semantic relation between the two entities and calculate the similarity of the two entity pairs, since that, give the answer to the unknown entity (entity “?”).

On a specific corpus, in general, Implicit Relational Search (IRS) consists of three main modules: The extracting module of the semantic relations from the corpus; Clustering module of semantic relations; Calculating module of similar relations between two entity pairs. In practice, the IRS model consists of two phases: online phase: meeting the real-time search, and offline phase: processing, calculating, storing semantic relations and similarity relations, in which, the extracting and clustering modules of the semantic relations are in the off-line phase of the IRS model.

*Extracting module of the semantic relations:* From the corpus, this module extracts the patterns as illustrated above:  $A$  the longest river  $B$ , where  $A, B$  are 2 named entities. The pattern set obtained will consist of different patterns, similar patterns, or patterns of different lengths and terms, but the same semantic expression. For example:  $A$  is the largest river of  $B$ ,  $A$  is the river of  $B$  has the largest basin, or  $B$  has the longest river as  $A$ , etc.

*Clustering module of semantic relations:* From the obtained pattern set, clustering is performed to identify clusters of contexts, where each context in the same clusters has a semantic similarity. Make a table of the pattern indexes and the corresponding entity pairs.

*Calculating module of similar relations between two entity pairs* is in the online phase of the IRS model. Pick up the query  $q = (A, B), (C, ?)$ , IRS will search the entity pair  $(A, B)$  and the corresponding semantic relation (context) set in the index table. From the obtained semantic relation set, find the pairs of entities  $(C, D_i)$  associated with this relation. Apply the Cosine measure to calculate and rank the similarity between  $(A, B)$  and  $(C, D_i)$ , and give a list of ranked entities  $D_i$  to answer the query.

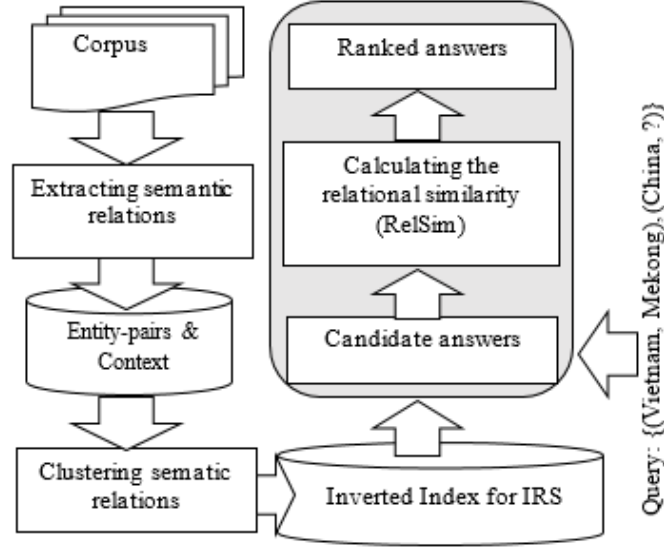


Figure 3. General structure of IRS

For illustration, provided that query  $q = (\text{Vietnam, Mekong}), (\text{China, ?})$ , IRS finds the cluster containing the entity pair (Vietnam, Mekong) and corresponding semantic relation: “the longest river” (from the source sentence: “Mekong is the longest river in Vietnam”). This cluster also contains a similar semantic relation: “the largest river”, in which “the largest river” ties with the entity pair (China, Yangtze) (from the source sentence: “Yangtze is the largest river in China”). The IRS will include “Yangtze” in the candidate list, rank the semantic relationship according to the similarity measurement, and return searching results.

In case IRS does not find  $A, B$  or  $C$ , the keyword search engine will be applied.

Receiving the input query  $q = (A, B), (C, ?)$ , the general structure of IRS is modeled

- Filter-Entities ( $F_e$ ) filters/seeks candidate set  $S$  containing entity pairs  $(C, D_i)$  that are related to the input entity pair  $(A, B)$

$$F_e(q, D) = F_e((A, B), (C, ?), D) = \begin{cases} 1, & \text{if Relevant}(q, D) \\ 0, & \text{else} \end{cases} \quad (1)$$

- Rank-Entities ( $R_e$ ) ranks the entities  $D_i, D_j$  in the candidate set  $S$  by RelSim (Relational Similarity), whichever has higher RelSim is ranked lower (i.e. closer with the top or higher rank)  
 $\forall D_i, D_j \in S$ , if

$$\text{RelSim}((A, B), (C, D_i)) > \text{RelSim}((A, B), (C, D_j)) : \text{Re}(q, D_i) < \text{Re}(q, D_j) \quad (2)$$

- List of  $L$  results from  $F_e$  and  $R_e$

$$\text{List } L = F_e \times R_e. \quad (3)$$

- Cosine measure (for  $x, y$  are the  $n$ -dimensional vectors)

$$\text{Cosine}(x, y) = \frac{\sum_{i=1}^n x_i \cdot y_i}{\sqrt{\sum_{i=1}^n x_i^2} \sqrt{\sum_{i=1}^n y_i^2}}. \quad (4)$$

### 3.2. Extracting module of semantic relations

Formally, the semantic relational extracting module performs contextual extraction of the surrounding/around (front, middle and behind) terms/patterns of the entity pairs. The context represents the semantic relationship of the entity pair.

Consider the sentence:  $a_1 a_2 \dots a_i A w_1 w_2 \dots w_j B b_1 b_2 \dots b_k$ .

For  $A, B$  are two entities, we have a context string  $p$  (context  $p$ ):

- Step 1. Set the condition and threshold:
  - Context must contain at least two entities  $A, B$ .
  - Context must contain at least 2 terms with a frequency  $\geq 10$  in the whole corpus.
- Step 2. Word segmentation, stop-word removal:
  - Use Longest-matching and dictionary (72721 terms) for word segmentation.
  - Use the Stop-word dictionary (2462 terms) to remove stop-words.
- Step 3. Recognition of NER, entity separation, pattern separation, calculation of TF, IDF, weight TF.IDF
  - NER is recognized by 9 general labels, Proper (uppercase letters), functions defining date formats, numerical types, rules that named entities combine: Mr., Ms., GS., TS., UBND, TCT, TNHH, etc.
  - After separating the NERs, the remaining context describes the semantics of the NERs.
  - Calculating TF (Term Frequency), IDF (Inverted Document Frequency), weight TF.IDF of terms. The calculation of weight TF.IDF is to eliminate terms in contexts that the weight is too low (noise, un-grammar) or too high (too common). The purpose is to narrow the search space, limit process steps and calculate Cosine.

The results of the extraction algorithm are saved in the database, illustrated as Figure 4: With two sentences: “Mỹ - Hàn phóng hàng loạt tên lửa dẫn mắt Triều Tiên” (“US - Korea launched a series of ballistic missiles to forewarn South Korea”); “Hiện nay xã Tân Tiến là một xã nông thôn mới tiêu biểu toàn tỉnh Hậu Giang” (“Tan Tien is a new typical rural commune over Hau Giang Province”).

NER	Context	TF	TF.IDF
Mỹ - Hàn_Triều Tiên (US - Korea_South Korea)	phóng_hàng_loạt_tên_lửa_dẫn_mặt (launch_a_series_of_ballistic missiles_forewarn)	0.25 0.25 0.25  0.25	2.8 5.31 3.77 5.61
Tân_Tiền_Hậu_Giang (Tan_Tien_Hau Giang)	xã_nông_thôn_mới_tiêu biểu_toàn_tỉnh (commune_rural_new_typical_over_ province)	0.17 0.17 0.17  0.17 0.17 0.17	2.28 5.61 3.32 5.61  2.87 3.22

Figure 4. Results of extraction algorithm

### 3.3. Clustering module of semantic relations

By default, when 2 pairs of entities are similar, their contexts will use the same terms. However, due to the natural expression of the language, two contexts with different terms can still be semantically similar. For example, consider two contexts: “Hoa Đà, thầy thuốc người Trung Hoa” (“HuaTuo, Chinese physician”) and “Tôn Thất Tùng, vị bác sĩ Việt Nam” (“Ton That Tung, Vietnamese doctor”), or two other contexts: “Lady Gaga, nữ ca sĩ người Mỹ” (“Lady Gaga, American singer”) and “Khánh Ly, người hát nhạc Việt” (“Khanh Ly, person singing Vietnamese songs”). Obviously, these contexts do not use common terms, but they are semantically similar. Thus, in order to identify semantic similarity between two contexts not using the same terms, the paper applies the Distributional Hypothesis.

The Distribution Hypothesis states that words used in the same context tend to have a semantic similarity. It means that, in a pattern set, where term  $t_1$  usually co-occurs with term  $t_2$  in each pattern, they have a high probability of semantic similarity [1].

Regarding the context level: In the corpus, if two contexts  $p_1, p_2$  often co-occur with the pair of entities  $(X, Y)$ , the two contexts  $p_1, p_2$  are semantically similar. For an example of two contexts: “ $X$  was acquired by  $Y$ ” and “ $X$  buys  $Y$ ” often co-occur with the entity pair (Google, Youtube) and/or (Adobe Systems, Macromedia), so 2 above contexts will be semantically similar [1]. On the mathematical basis, this extended hypothesis is similar to the vector space model. Considering each context as a weighted vector, if the more pairs of entities the two contexts co-occur in, the higher Cosine similarity between the two vectors is.

The extension of Distribution Hypothesis has been successfully applied in the studies identifying the semantic similarity, or improved the accuracy in semantic similarity measurement [1]. In the studies [9, 1, 10], the authors made a mathematization of the extension of the distribution hypothesis according to the formulas

- The denotation of the entity pair  $(A, B)$  is  $w$ , in the whole corpus, we consider  $P(w)$  as the set of contexts where  $w$  co-occurs

$$P(w) = p_1, p_2, \dots, p_n. \quad (5)$$

- Similarly,  $W(p)$  is the set of pairs of entities that the context  $p$  co-occurs

$$W(p) = w_1, w_2, \dots, w_m. \quad (6)$$



- Consider the frequency of co-occurrence of  $w_i$  and  $p_j$  in the same context sentence as  $f(w_i, p_j)$ . The frequency vector of pairs of entities  $\phi(p)$  where the context  $p$  co-occurs is defined

$$\phi(p) = (f(w_1, p), f(w_2, p), \dots, f(w_m, p))^T. \quad (7)$$

- Similarly, the frequency vector  $\phi(w)$  of context set co-occurs with the pair of entities  $w$

$$\phi(w) = (f(w, p_1), f(w, p_2), \dots, f(w, p_n))^T. \quad (8)$$

- The measure similarity between 2 contexts  $p, q$

$$\text{Sim}_{DH}(p, q) = \text{Cosine}(\phi(p), \phi(q)) = \frac{\sum_i (f(w_i, p) \cdot f(w_i, q))}{\|f(w_i, p)\| \|f(w_i, q)\|}. \quad (9)$$

- $\|f(w, p)\|$  is  $L_2$ -norm

$$\|f(w, p)\| = \sqrt{\sum_i (f^2(w, p))}. \quad (10)$$

However, the distribution hypothesis requires pairs of common entities always to “co-occur” with the context. Measuring similarity based on distribution hypothesis, not on term similarity, will significantly affect the quality of clustering techniques. Therefore, in the clustering algorithm, the paper proposes the combination between the distributional hypothesis and the Cosine measure. The combination is described as follows:

- Term based similarity measure of two contexts  $p, q$

$$\text{Sim}_{term}(p, q) = \frac{\sum_{i=1}^n (\text{weight}_i(p) \times \text{weight}_i(q))}{\|\text{weight}(p)\| \|\text{weight}(q)\|}. \quad (11)$$

- Combined similarity measure

$$\text{Sim}(p, q) = \max(\text{Sim}_{DH}(p, q), \text{Sim}_{term}(p, q)). \quad (12)$$

- Each cluster  $C$  consists of a set of similar contexts  $p_i$ , the cluster center is vectorized [7]

$$\text{Centroid } \vec{C} = \text{norm} \left( \frac{\sum_{p_i \in C} \vec{p}_i}{|C|} \right) \quad (13)$$

- Clustering algorithm (improved from [1, 10] :

- Input: Set  $P = p_1, p_2, \dots, p_n$ ; Clustering threshold  $\theta$ , heuristic threshold  $\theta_1$ ; Cluster diameter  $Dmax$ .
- Output: A set of resulting clusters Cset (containing the Cluster\_ID, context, weight for each context, and corresponding entity pair).
- Each cluster contains similarly semantic contexts.

```

program Clustering_algorithm
// Initialize the cluster set with empty.
01.  Cset = {}
02.  for each pattern  $p_i \in P$  do
03.    Dmax = 0;  $c^* = \text{NULL}$ ;
04.    for each cluster  $c_j \in \text{Cset}$  do
05.      Sim_cp=Sim( $p_i$ , Centroid( $c_j$ ))
06.      if (Sim_cp > Dmax) then
07.        Dmax = Sim_cp;  $c^* \rightarrow c_j$ ;
08.      end if
09.    end for
10.    if (Dmax >  $\theta$ ) then
11.       $c^*.append(p_i)$ 
12.    else
13.      Cset  $\cup = c^*$ 
14.    end if
15.    if ( $i > \theta 1$ ) then
16.      exit Current_Proc_Cluster_Algo()
17.    end if
18.  end for
@CallMerge_Cset_from_OtherNodes()
end.

```

Interpretation: The clustering threshold  $\theta \in [0, 1]$  is entered into set of context  $P$ . Threshold  $\theta 1$  from line 15-19 is a heuristic for other nodes to perform concurrently when clustering with a large number of contexts (over 400000). The global array Cset is used to compare, mix, and de-duplicate of the resulting Cluster\_ID.

In the algorithm, the function  $\text{Sim}(p_i, \text{Centroid}(c_j))$  applies the combined similarity measure that the paper proposed. The loop body from line 5 to line 8, looks up the cluster  $c_j$  that is most similar to the context  $p_i$  being considered. If the value of the function is greater than the clustering threshold  $\theta$ , add context  $p_i$  to the current cluster  $c_j$ , otherwise, a new cluster containing only  $p_i$  is generated (sequence of statements 10-14). Similarity calculations of cluster centers are applied according to the formulas (10) - (14).

The algorithm has a computational complexity of  $O(N.C)$ , where  $N$  is the number of contexts to cluster,  $C$  is the total number of clusters ( $N \gg C$ ). Applying heuristic parallelism, the algorithm can be clustered in a relatively short time, even when context set is very large.

Applying the Distributional Hypothesis can overcome the disadvantages of two contexts that can be semantically similar even if there are no common terms. Applying the term based similarity measure does not require that the common pairs of entities must usually “co-occur” with contexts. Therefore, the study combines the function of term based similarity measure and Distributional Hypothesis in the clustering step. Experiments show that the quality of clusters is significantly improved when the above combination is applied.

### 3.4. Modules calculating the relational similarity between two pairs of entities

It is included in the online phase of IRS model. This calculation is considered as a difficult problem because: Firstly, the similarity in relationships changes from time to time. For example, consider the two entity pairs (Trump, the current US President) and (Elizabeth, the Queen of England), the relational similarity will change over different terms. Secondly, each entity pair appears in many different semantic relationships, for example: “Russia successfully held the World Cup 2018”; “Russia prevents terrorism during the World Cup”; “Russia reaches the World Cup 2018 quarterfinals”, etc. Thirdly, the notion similarity between entity pairs can be illustrated by more than one way, for example, “ $X$  was acquired by  $Y$ ” and “ $X$  buys  $Y$ ”. Fourthly, it is complex when an entity has a name within itself (person, organization, place names, etc.) which are not common ones or not listed in dictionaries. Finally, entity  $D$  is unknown and it is in the searching phase.

The module calculating the relational similarity between two pairs of entities execute two tasks: Filtering (searching) and ranking. As illustrated in Subsection 3.1, the input query  $q = (A, B), (C, ?)$ , through the inverted index, IRS executes the function Filter-Entities  $Fe$  to filter (search) out candidate sets having entity pairs  $(C, D_i)$  and the corresponding context, such that  $(C, D_i)$  similar to  $(A, B)$ . Then, it executes the function Rank-Entities  $Re$  to rank the entities  $D_i, D_j$  within the candidate set according to RelSim measure (Relational Similarity), finally - which results in the list of ranked  $D_i$ .

The filtering context relationships process is carried out with limiting conditions: The context must contain at least two entities  $A, B$ , and at least two terms, each term comes with a frequency not less 10 within the whole corpus. The limiting conditions have narrowed down a significant area of searching. The Cset search results set includes: ClusterID, context, context weight, and the corresponding entity pair - stored as inverted index. The function Filter-Entities is basically a list of context and entities which meet the conditions belonging to  $(A, B)$ . This function does not require any similarity computation, the for loop only proceeds in a very small subset, the number of context is similar to the input  $(A, B)$  pair. Thus, the computing time of Filter-Entities function is a constant, suitable for real-time filtering.

- The function Filter-Entities:
  - Input: Query  $q = (A, B), (C, ?)$ .
  - Output: The candidate set  $S$  (includes entities  $D_i$  and its corresponding context).

#### program Filter\_entities

```

01.  Cset = {}
02.  P(w) = EntPairfromCset.Context();
03.  for each context  $p_i \in P(A, B)$  do
04.      W(p)=Context( $p_i$ ).EntPairs();
05.      S  $\cup$  = W(p);
06.  end for
07.  return S;
```

end.

Although, Filter-Entities can filter out the candidate set having some answers, it cannot ensure that the candidate entities  $D_i$  in the target entity pair  $(C, D_i)$  is accurate and con-

sistent semantically with the source entity pair  $(A, B)$ . For example, with the query  $q = (\text{Ly Thuong Kiet, Nhu Nguyet}, (\text{Napoléon, ?}))$ , if the semantic relationship is: “famous battle”, the answer is “Waterloo”. However, in corpus, the semantic relationship of  $(\text{Ly Thuong Kiet, Nhu Nguyet})$  pair has such models: “Defense line”, “campaign”, etc. When we consider the context “campaign” separately, Napoléon has about 7 alliance campaigns, IRS could not return the most appropriate answer. Therefore, IRS model executes RelSim measure (a Cosine variant) to rank entities  $D_i, D_j$  within the candidate set and returns the list of ranked  $D_i$  entities as the results.

After executing Filter-Entities, a subset of the entities  $D_i$  and corresponding context are obtained. RelSim only processes and calculates on the very small subset. In addition, RelSim uses the threshold  $\sigma$  to eliminate entities  $D_i$  with low RelSim values.

For  $F_e(q, D) = F_e((A, B), (C, ?), D)$

$$F_e(q, D_i) = \begin{cases} 1, & \text{if RelSim}((A, B), (C, D_i)) > \alpha \\ 0, & \text{else.} \end{cases} \quad (14)$$

- The Rank-Entities algorithm.

Inputs include:

- Source entity-pair  $(A, B)$ , denoted by  $s$ ; Candidate entities  $(C, D_i)$ , denoted by  $c$ ;
- Contexts correspond to  $s, c$ ;
- For known entities  $A, B$ , and  $C \leftarrow$  the corresponding cluster set containing  $A, B$ , and  $C$  is identified;
- Threshold  $\sigma$  (compared to RelSim value); Threshold  $\sigma$  is set during testing the program.
- Cluster set: Cset; Initialize Dot-product ( $\beta$ ); set used-context ( $\theta$ );  
Output: List of answers (List of ranked entities)  $D_i$ .
- Denotations:  $P(s), P(c)$  given in formula (6);
- $f(s, p_i), f(c, p_i), \phi(s), \phi(c)$  given in (8), (9);
- $\gamma$ : Variable (set of context) keeps the given context;
- $q$ : Temporary variable (Context);  $\omega$ : Cluster;

#### program Rank\_Entities

01. for each context  $p_i \in P(c)$  do
02.   if  $(p_i \in P(s))$  then
03.      $\beta \rightarrow \beta + f(s, p_i).f(c, p_i)$
04.      $\gamma \rightarrow \gamma \cup p$
05.   else
06.      $\omega \rightarrow$  cluster contains  $p_i$
07.      $\max\_co\text{-occurs} = 0$ ;
08.      $q \rightarrow \text{NULL}$

```

09.   for each context  $p_j \in (P(s) \setminus P(c) \setminus \gamma)$  do
10.       if  $(p_j \in \omega) \ \& \ (f(s, p_j) > \max\_co\text{-occurs})$ 
11.            $\max\_co\text{-occurs} \rightarrow f(s, p_j)$ ;
12.            $q \rightarrow p_j$ ;
13.       end if
14.   end for
15.   if  $(\max\_co\text{-occurs} > 0)$  then
16.        $\beta \rightarrow \beta + f(s, q) \cdot f(c, p_i)$ 
17.        $\gamma \rightarrow \gamma \cup q$ 
18.   end if
19. end if
20. end for
21.  $RelSim \rightarrow \beta / L2 - \text{norm}(\phi(s), \phi(c))$ 
22. if  $(RelSim \geq \alpha)$  then return RelSim
end.

```

Interpretation: In the case two pairs of source and target entities have the same semantic relationship (sharing the same context, statement 1-2)  $p_i \in P(s) \cap P(c)$ , calculate the dot product as a modified version of standard Cosine similarity formula.

In the case of  $p_i \in P(c)$  but  $p_i \notin P(s)$ , the algorithm finds the context  $p_j$  (or temporary variable,  $q$ , line 12), where  $p_i, p_j$  belong to the same cluster. The loop body (from statements 10-13) chooses the context  $p_j$  has largest frequency of co-occurrence with the  $s$ . Under the distribution hypothesis, the more pairs of entities two contexts  $p_i, p_j$  co-occur in, the higher Cosine similarity between the two vectors. As the cosine value is higher,  $p_i, p_j$  are more similar. Therefore, the pair  $(C, D_i)$  is more accurate and semantically consistent with the source entity pair  $(A, B)$ .

The sequence of statements from 15-18 calculates the dot product. Statements from 21-22 calculate the RelSim value. From the set of RelSim value, whichever entities  $D_i$  have RelSim higher will be ranked lower (in the closer top, or higher rank). Finally, the result set  $D_i$  is the answer list for the query that the end-user wants to find.

## 4. EXPERIMENT - CONCLUSIONS

### 4.1. Experiment

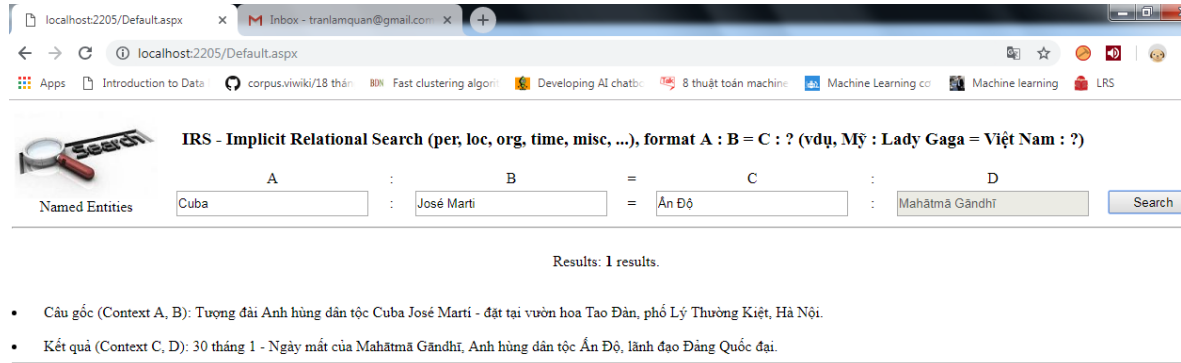
The dataset was downloaded from Viwiki (7877 files) and Vn-news (35440 files) because these datasets consist of patterns containing Named Entity. After reading, extracting file content, separating paragraphs and sentences (main-sentences, sub-sentences), 1572616 sentences are obtained.

The paper selects the classification of Named Entity Recognition (NER) at the general level, with 9 labels: B-LOC; B-MISC; B-ORG; B-PER; I-LOC; I-MISC; I-ORG; I-PER and O, where the prefix B-Begin and I-Inner correspond to the start and inner positions of the entity name.

NER general labels: PER: Name of person; ORG: Organization Name; LOC: Place Name; TIME: Time type; NUM: Number mode; CUR: Currency type; PCT: Percentage type; MISC: Other entity types; O: Not an entity.

By using the algorithm for extracting patterns saved in the database, after performing the processing steps and the limit conditions, there are 404507 context sentences left in the database.

As far as we know, in Vietnamese, there is no implicit semantic relational entity search model. Therefore, the paper has no conditions for comparing and testing with other methods.



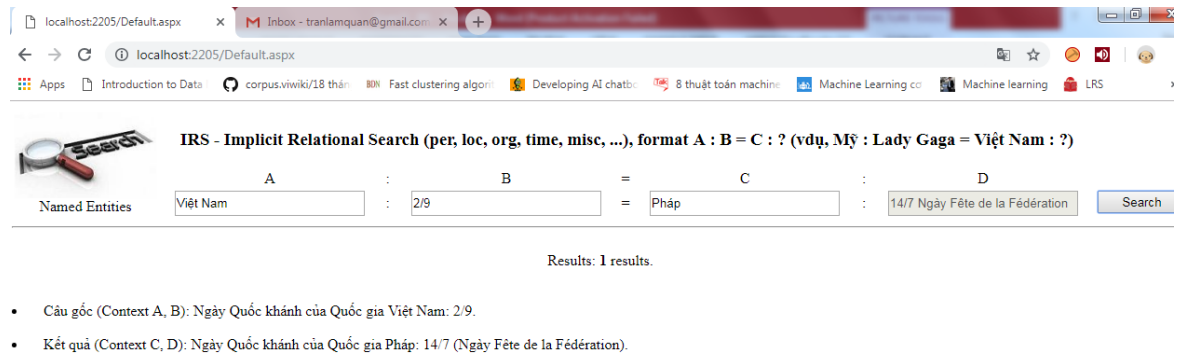
IRS - Implicit Relational Search (per, loc, org, time, misc, ...), format A : B = C : ? (vd, Mỹ : Lady Gaga = Việt Nam : ?)

Named Entities: Cuba : José Martí = An Đô : Mahatma Gandhi

Results: 1 results.

- Câu gốc (Context A, B): Tượng đài Anh hùng dân tộc Cuba José Martí - đặt tại vườn hoa Tao Đàn, phố Lý Thường Kiệt, Hà Nội.
- Kết quả (Context C, D): 30 tháng 1 - Ngày mất của Mahatma Gandhi, Anh hùng dân tộc Ấn Độ, lãnh đạo Đảng Quốc đại.

Figure 5. Experiment of IRS with B-PER entity label



IRS - Implicit Relational Search (per, loc, org, time, misc, ...), format A : B = C : ? (vd, Mỹ : Lady Gaga = Việt Nam : ?)

Named Entities: Việt Nam : 2/9 = Pháp : 14/7 Ngày Fête de la Fédération

Results: 1 results.

- Câu gốc (Context A, B): Ngày Quốc khánh của Quốc gia Việt Nam: 2/9.
- Kết quả (Context C, D): Ngày Quốc khánh của Quốc gia Pháp: 14/7 (Ngày Fête de la Fédération).

Figure 6. Experiment of IRS with the entity of time type

In order to evaluate accuracy, we performed 500 queries (in the form of  $((A : B)(C : ?))$  and after testing, the result's accuracy is approximately 90%.

Some exceptions need to be solved within the scope of research, when users put into a query that is random, arbitrary and does not follow the motive of  $q = (A, B), (C, ?)$ . In the case where the semantic relationship of the input pair of entities  $(A, B)$  is 0, in other words, in the corpus, semantic relations  $(A, B)$  or even the pair of entities  $(A, B)$  does not exist, the IRS model will transform into a search mechanism according to keywords and context-aware [7]. In another case, the query is under the motive  $q = (A, B), (C, ?)$ , however, in fact, the relation of the pair of entities  $(A, B)$  is not only mono-semantic but can be poly-semantic. At that time, there will be many different semantic relations in the same pair of entities.

ID	A	B	C	D
..	German	Angela Merkel	Israel	Benjamin Netanyahu
..	Harry Kane	Tottenham	Messi	Barca
..	Chí Anh	Khánh Thi	Khắc Việt	Tuấn Hưng
..	...	...	...	...
..	Hà Nội	Kim Quy	Hoàn Kiếm	Gươm Thần Kim Quy (Magical sword – Golden Turtle)
..	Hoàng Công Lương	Hòa Bình	Thiên Sơn	RO

Figure 7. Example for experimental results with input  $q = A, B, C$  and output

For example, the pair of entities (Notre Dame: Paris) will have semantic relations such as “a fire”, “a symbol”, “a literary work”, “a love story of Hunchback”, “a crown of thorns”, etc. With known  $C$  entity, Rank-Entities algorithm seeks candidate clusters, returning the result as a list of  $D_i$  entities that are similarity maximized to the semantic relation of the pair of entities ( $A, B$ ).

#### 4.2. Conclusions

The ability to infer undefined information/knowledge by inference is one of the natural abilities of human. The paper presents an Implicit Relational entity search model (IRS) that simulates the above ability. The IRS model searches for information/knowledge from an unfamiliar semantic domain and does not require keywords known in advance, using the same example (similar relation) from a familiar semantic domain. The contributions in the paper include the researches and implementation of the IRS system on the Vietnamese language domain, applied to 9 classes of entities, proposing a similarity measure in combination between terms and distributional hypothesis; thus, applied the heuristic to the clustering algorithm to improve cluster quality.

In terms of direction of the paper, on the one hand, the paper considers more types of relational mapping and time addition elements for the search results to be updated and more accurate. On the other hand, it is possible to extend entity search with an input query that includes only one entity, for example, “What is the longest river in China?”, the implicit relational search model will give the correct answer: “Truong Giang” although Corpus only contains the source sentence “Truong Giang is the largest river in China”.

#### REFERENCES

- [1] D. T. Bollegala, Y. Matsuo, and M. Ishizuka, “Measuring the similarity between implicit semantic relations from the web,” in *Proceedings of the 18th International Conference on World Wide Web*, ser. WWW '09. New York, NY, USA: ACM, 2009, pp. 651–660. [Online]. Available: <http://doi.acm.org/10.1145/1526709.1526797>
- [2] D. Gentner, “Structure-mapping: A theoretical framework for analogy\*,” *Cognitive Science*, vol. 7, pp. 155–170, April 1983.
- [3] J. Guo, G. Xu, X. Cheng, and H. Li, “Named entity recognition in query,” in *Proceedings of the 32Nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '09. New York, NY, USA: ACM, 2009, pp. 267–274. [Online]. Available: <http://doi.acm.org/10.1145/1571941.1571989>

- [4] B. Hjørland, “Link: <http://vnlp.net>.”
- [5] J. Jansen and A. Spink, “How are we searching the world wide web? a comparison of nine search engine transaction logs,” *Information Processing & Management*, vol. 42, pp. 248–263, 01 2006.
- [6] Y. Jiao Cao, Z. Lu, and S. Mei Cai, “Relational similarity measure: An approach combining wikipedia and wordnet,” *Applied Mechanics and Materials*, vol. 55-57, 05 2011.
- [7] T. Lam Quan and V. Tat Thang, “A study of applying vietnamese voice interaction for a context-based aviation search engine,” in *The IEEE RIVF International Conference on Computing and Communication Technologies*, 2013.
- [8] M. P. Kato, H. Ohshima, S. Oyama, and K. Tanaka, “Query by analogical example: Relational search using web search engine indices,” 01 2009, pp. 27–36. [Online]. Available: doi:10.1145/1645953.1645960
- [9] M.-T. Pham, “Principal,” Ph.D. dissertation. [Online]. Available: [https://www.researchgate.net/publication/221604624\\_Cross-Language\\_Latent\\_Relational\\_Search\\_Mapping\\_Knowledge\\_across\\_Languages](https://www.researchgate.net/publication/221604624_Cross-Language_Latent_Relational_Search_Mapping_Knowledge_across_Languages)
- [10] N. Tuan Duc, D. Bollegala, and M. Ishizuka, “Cross-language latent relational search: Mapping knowledge across languages,” in *In Proceedings of the International Conference on Recent Advances in Natural Language Processing RANLP, 2005*, vol. 2, 01 2011. [Online]. Available: [https://www.researchgate.net/publication/1957756\\_Combining\\_Independent\\_Modules\\_in\\_Lexical\\_Multiple-Choice\\_Problems](https://www.researchgate.net/publication/1957756_Combining_Independent_Modules_in_Lexical_Multiple-Choice_Problems)
- [11] P. Turney, M. Littman, J. Bigham, and V. Shnayder, “Combining independent modules in lexical multiple-choice problems,” *Recent Advances in Natural Language Processing III: Selected papers from RANLP 2003*, 2004. [Online]. Available: doi:10.1075/cilt.260.11tur
- [12] P. D. Turney, “Similarity of semantic relations,” *Comput. Linguist.*, vol. 32, no. 3, pp. 379–416, Sep. 2006. [Online]. Available: <http://dx.doi.org/10.1162/coli.2006.32.3.379>
- [13] T. Veale, “Wordnet sits the s.a.t. a knowledge-based approach to lexical analogy,” in *Proceedings of the 16th European Conference on Artificial Intelligence*, ser. ECAI’04. Amsterdam, The Netherlands, The Netherlands: IOS Press, 2004, pp. 606–610. [Online]. Available: <http://dl.acm.org/citation.cfm?id=3000001.3000128>

*Received on October 19, 2018*

*Revised on April 24, 2019*