



- (51) **International Patent Classification:**
G06F 17/27 (2006.01)
- (21) **International Application Number:**
PCT/MY2015/050025
- (22) **International Filing Date:**
22 April 2015 (22.04.2015)
- (25) **Filing Language:** English
- (26) **Publication Language:** English
- (30) **Priority Data:**
PI2014701129 5 May 2014 (05.05.2014) MY
- (71) **Applicant:** MIMOS BERHAD [MY/MY]; Technology Park Malaysia, Kuala Lumpur, 57000 (MY).
- (72) **Inventors:** CHU, Benjamin Min Xian; c/o Mimos Berhad, Technology Park Malaysia, Kuala Lumpur, 57000 (MY). QIANG, Simon Liu; c/o Mimos Berhad, Technology Park Malaysia, Kuala Lumpur, 57000 (MY). LUKOSE, Dickson; c/o Mimos Berhad, Technology Park Malaysia, Kuala Lumpur, 57000 (MY).
- (74) **Agent:** H A RASHID, Ahmad Fadzlee; Ipvolsi Sdn Bhd, A-3-3a, Centrio Pantai Hillpark, No. 1, Jalan Pantai Murni, Kuala Lumpur, 59200 (MY).
- (81) **Designated States** (*unless otherwise indicated, for every kind of national protection available*): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.
- (84) **Designated States** (*unless otherwise indicated, for every kind of regional protection available*): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK,

[Continued on next page]

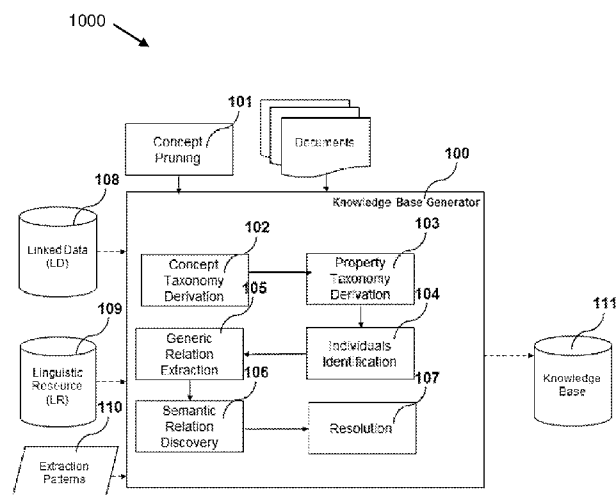
(54) **Title:** SYSTEM AND METHOD FOR AUTOMATICALLY GENERATING A KNOWLEDGE BASE

FIG. 1

(57) **Abstract:** The present invention relates to a system (1000) and method for generating knowledge base (111) automatically from structured and unstructured texts. The system (1000) which generates a knowledge base (111) automatically from natural language texts, comprises of a Knowledge Base Generator (100) which utilises a Linked Data (LD) (108); a Linguistic Resource (LR) (109); and an Extraction Patterns component (110) to support a series of process and techniques for the Knowledge Base Generator (100). The Knowledge Base Generator (100) further comprises a Concept Pruning component (101), a Concept Taxonomy Derivation component (102), a Property Taxonomy Derivation component (103), an Individuals Identification component (104), a Generic Relation Extraction component (105), a Semantic Relation Discovery component (106) and a Resolution component (107).



SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG). **Published:**

Declarations under Rule 4.17:

- *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))*
- *of inventorship (Rule 4.17(iv))*

- *with international search report (Art. 21(3))*
- *before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments (Rule 48.2(h))*

SYSTEM AND METHOD FOR AUTOMATICALLY GENERATING A KNOWLEDGE BASE

FIELD OF INVENTION

5 The present invention relates to a system and method for generating knowledge base automatically. More particularly, the present invention relates to a system and method for generating knowledge base automatically from structured and unstructured texts.

10 BACKGROUND OF THE INVENTION

 A knowledge base is a special database for knowledge management. It is an information repository that provides a means for information to be collected, organised, shared, searched and utilized. A knowledge base can either be machine-readable or intended for human use. Human-readable knowledge bases are used to
15 enable people to retrieve and use the knowledge they contain. A human-readable knowledge base can also be coupled with a machine-readable one. Machine-readable knowledge bases store knowledge in a computer-readable form which usually has automated deductive reasoning applied to them.

20 These knowledge bases contain a set of data, often in the form of rules that describe the knowledge in a logically consistent manner, whereby an ontology is used to define the structure of the stored data. Knowledge base generation is a complex task as it is laborious and is usually manual. It needs to be updated and enriched continuously. Besides that, the mining-based approach implemented in
25 natural language processing (NLP) techniques still requires human assistance.

 An example of a method for building knowledge structures consisting of concepts and interrelations and representing knowledge extracted from the input text is disclosed in the International Patent Publication Number WO 2005/062202.

30 The method involves reading an input, using semantic based means for extracting concepts and the interrelations from said input text, transforming said input text into a machine understandable knowledge representation so as to provide knowledge libraries from said documents and optionally storing said libraries.

However, the common existing system and method for managing or generating a knowledge base is laborious and troublesome as it continuously needs to be updated and enriched. There is no disclosure of the techniques of generating a knowledge base automatically from both structured and unstructured collection of texts. Therefore, there is a need to provide a system and method that addresses the above mentioned drawbacks of the existing system and method for knowledge base generation when both entities and relations are given.

SUMMARY OF INVENTION

The present invention relates to a system and method for generating knowledge base automatically from structured and unstructured texts. The system (1000) for generating knowledge base (111) automatically comprises of a Knowledge Base Generator (100); a Linked Data (LD) (108); a Linguistic Resources (LR) (109); and an Extraction Patterns component (110) to support a series of process and techniques for the Knowledge Base Generator (100). Additionally, the Knowledge Base Generator (100) includes a Concept Pruning component (101) to identify salient entities or the right concepts to be included; a Concept Taxonomy Derivation component (102) to identify how a particular concept relates to its parent concept by referring to Linked Data (108) or Linguistic Resource (109); a Property Taxonomy Derivation component (103) to arrange all derived relations into a hierarchy; an Individuals Identification component (104) to detect individual and its concept type that is involved in a sentence; a Generic Relation Extraction component (105) to extract generic relations which can be discovered from Linguistic Resource (109) or the Extraction Patterns (110); a Semantic Relation Discovery component (106) to discover implicit relations that the Generic Relation Extraction component (105) is not able to discover; and a Resolution component (107) to resolve a best relation candidate that can be applied to a context of a text as there are many possible relation candidates.

The method for generating knowledge base (111) automatically is characterised by the steps of tokenising a paragraph into sentences; identifying if a sentence has an appositive phrase type by a Generic Relation Extraction component (105); identifying if the appositive phrases are essential or non-essential; referring to a vector list to determine the current position for each of the appositive phrases; retrieving the entity prior to the appositive phrase; identifying all salient entities from

sentences to extract generic relations; performing a check on all the pairs; and adding possible pairs which represent the relations of a particular concept to another concept and to a knowledge base (111).

5 Preferably, the step of performing a check on all the pairs includes determining if either one of the entities or the prior entity of the pairs is a name; identifying the entity as an "*instance*" if the entity is a name; and adding the "*instance*" into an individual index.

10 Preferably, if the entity is not identified as an "*instance*", the steps include determining if either one of the entities or the prior entity is an animate; and identifying the entity with a generic relation of "*is-a*" between subject and object of the entity if the entity is an animate.

15 Preferably, if either one of the entities or the prior entity is not an animate the entity is identified with a generic relation of "*hasType*" between subject and object of the entity.

20 Preferably, if a sentence is identified as a non-appositive phrase type, the steps include retrieving all salient entities that are identified by the Generic Relation Extraction component (105), wherein these entities are managed by a vector list; selecting each entry in the vector list and checking if the entities exist in a knowledge base (111); checking the entity if it is defined in a Linguistic Resource (109) or a Linked Data (108) if the entity is not in the knowledge base (111); selecting a pattern
25 entry if the pattern can be applied, wherein each pattern has a corresponding property type; and adding a triple to the knowledge base (111) if the pattern can be applied.

30 Preferably, if there is no pattern can be applied, the steps include extracting sense descriptions of the salient entity from the Linguistic Resource (109); transforming the sense description into context semantic structures; retrieving all context semantic structures from context index in selecting the best candidate by a Resolution component (107); performing graph similarity matching between identified sentence and the context semantic structures; and choosing and adding the best
35 candidate triple information into the knowledge base (111)

Preferably, the step of choosing and adding the best candidate triple information includes retrieving all context semantic structures from the context index by the Resolution component (107); performing semantic graph matching for each of the context semantic structures; identifying all sub graphs attached to the verb for each of the candidate context semantic structures; checking if the concept conforms to the semantic constraint for each of the concepts; merging the concept to the semantic structure once the concept conforms to the semantic constraint to produce a new sub graph; adding a sub graph count with a value of 1; consolidating all the sub graph counts; merging all the sub graphs once the iteration is complete; selecting a top context entry with the highest total sub graph counts; and adding the best candidate triple information into the knowledge base.

Preferably, if the entity is found in the Linked Data (109), the steps include extracting an abstract description that defines the entity; and transforming the abstract description into context semantic structure.

BRIEF DESCRIPTION OF THE DRAWINGS

The accompanying drawings, which are incorporated in and constitute a part of the specification, illustrate embodiments of the invention and, together with the description, serve to explain the principles of the invention.

FIG. 1 illustrates a system (1000) for generating knowledge base automatically according to an embodiment of the present invention.

FIGS. 2 (a) – (c) illustrate flow charts for automated knowledge base generation process according to an embodiment of the present invention.

FIG. 3 (a) illustrates an example of a whole graph structure.

FIGS. 3 (b – d) illustrate sub graphs examples which are only a part of the whole graph structure of FIG. 3 (a).

DESCRIPTION OF THE PREFERRED EMBODIMENT

A preferred embodiment of the present invention will be described herein below with reference to the accompanying drawings. In the following description, well known functions or constructions are not described in detail since they would obscure the description with unnecessary detail.

Reference is made initially to **FIG. 1** which illustrates a system (**1000**) for generating knowledge base (**111**) automatically according to an embodiment of the present invention. The system (**1000**) which generates a knowledge base (**111**) automatically from natural language texts, comprises of a Knowledge Base Generator (**100**) which utilises external resources such as Linked Data (LD) (**108**) Linguistic Resource (LR) (**109**) and an Extraction Patterns component (**110**) to support a series of process and techniques for the Knowledge Base Generator (**100**). The Knowledge Base Generator (**100**) further comprises a Concept Pruning component (**101**), a Concept Taxonomy Derivation component (**102**), a Property Taxonomy Derivation component (**103**), an Individuals Identification component (**104**), a Generic Relation Extraction component (**105**), a Semantic Relation Discovery component (**106**) and a Resolution component (**107**).

The Concept Pruning component (**101**) is used to identify salient entities or the right concepts to be included by filtering all the irrelevant ones. The Concept Taxonomy Derivation component (**102**) is used to identify how a particular concept relates to its parent concept by referring to Linked Data (**108**) or Linguistic Resource (**109**). It then eventually builds up the whole taxonomy for all the concepts. The Property Taxonomy Derivation component (**103**) is then used to arrange all derived relations into a hierarchy. The Individuals Identification component (**104**) is used to detect the individual and concept type in the sentence that is said to be appositive. The Generic Relation Extraction component (**105**) is used to extract generic relations discovered from Linguistic Resource (**109**) or from the Extraction Patterns component (**110**). Linguistic Resource (**109**) comprises the standard relations of linking of a particular concept, wherein the relations are considered as taxonomic type of relations. The Semantic Relation Discovery component (**106**) is used to discover implicit relations that Generic Relation Extraction component (**105**) is not able to discover. This can be done by referring to Linked Data (**108**) to know how a particular concept relates to another concept, wherein these relations are considered

as non-taxonomic type of relations. Finally, the Resolution component (107) is used to resolve the best relation candidate that can be applied to a context of the text as there are many possible relation candidates.

5 Referring to FIGS. 2 (a - c), there are illustrated flow charts for a method for generating knowledge base automatically according to an embodiment of the present invention. Initially, any tokenisation method is used to tokenize a paragraph into sentences, wherein these sentences are stored into a sentence vector. These sentences are fetched from the sentence vector to be processed. Once all the
10 sentences are processed as in decision 201, entities of the sentences are checked if the entities are defined in the Linguistic Resource (109) or Linked Data (108) as in decision 217 shown in FIG. 2 (b).

However, if there are still sentences not processed as in decision 201, the
15 Generic Relation Extraction component (105) selects the next sentence from the sentence vector which also stores the semantic structure that represents each sentence and the type of phrase of the sentence as in step 202. The Generic Relation Extraction component (105) then identifies if the sentence has an appositive phrase type as in step 203. These appositive phrases can be defined as essential or
20 non-essential, wherein essential appositives are not set off with commas, and wherein non-essential appositives are defined with the use of commas.

An appositive phrase is said to be essential when a sentence is made clear without the use of commas. For example, in a sentence "*The author J.K. Rowling is a*
25 *British novelist*," wherein "*J.K.Rowling*" is referred as appositive, the sentence is said to be essential as a comma is not used to make it clear. On the other hand, a non-essential appositive phrase is when commas are used to make the meaning of a sentence clear. An example of a non-essential appositive phase is "*My teacher, Miss Marshall is teaching us history*," wherein the sentence uses commas to make the
30 meaning of the sentence clearer.

If appositive phrases are found, the Generic Relation Extraction component (105) refers to a vector list of $E_1, E_2, \dots, E_n$ to determine the current position for each of the appositive phrase and retrieves the entity prior to it. For example, if the
35 appositive phrase found is E_2 , the Generic Relation Extraction component (105) will

retrieve E_1 as in step 204. The Generic Relation Extraction component (105) then identifies all salient entities from sentences to extract generic relations using any recognition method as in step 212 and these entities are stored and managed by the vector list as in step 213. An example of a vector list is $V1 = \{\textit{tennis-player},$
5 $\textit{tournament}, \textit{ball}, \textit{grand-slam}\}$, wherein *tennis-player*, *tournament*, *ball* and *grand-slam* are salient entities that are extracted from a text.

It then iterates all the entities in the vector list until it completes for one appositive phrase. Once it has completed iterating for one appositive phrase, the
10 Generic Relation Extraction component (105) continues with the next appositive phrase. All appositive phrases are processed and all possible pairs are identified to be checked by the Generic Relation Extraction component (105) as in steps 205 and 206. These possible pairs which represent the relations of a particular concept to another concept and will finally be added to the knowledge base (111) as in steps
15 210 and 211.

Before these possible pairs are being added to the knowledge base (111), these possible pairs first need to be checked to determine if either one of the entities or the prior entity of the pairs is a name that refers to a person or an organisation as
20 in decision 207. If it is, the entity is then identified as an “*instance*” and the “*instance*” is then added into the individual index as in step 208. An individual index is a list of index to manage all individuals. Examples of individual index are *Roger Federer*, *tennis-player*, *Joshua Bell*, *violinist*, and *Isaac Newton*, *mathematician*. On the other hand, if the entity is not identified as an “*instance*” as in decision 207, the Generic
25 Extraction component (105) then directly checks if either one of the entities or its prior entity is an animate as in decision 209 without adding anything into the individual index. If either one of the entities or its prior entity is an animate, the entity is identified with a generic relation of “*is-a*” between the subject and object as in step 210.

30

For example, “*My uncle, a lawyer is visiting us.*” From this sentence, the salient entities are identified as “*uncle*” and “*lawyer.*” The entity “*uncle*” is identified as an “*instance*” and is referred as an animate concept by the Generic Relation Extraction component (105). Therefore, a triple with the generic relation of “*is-a*” is

put between the subject “*uncle*” and object “*/lawyer*.” Hence, a triple of <*uncle*, *is-a*, */lawyer*> is added into the knowledge base (111).

However, if either one of the entities or its prior entity is not an animate as in decision 209, the entity is then identified with a generic relation of “*hasType*” between the subject and object as in step 211.

For example, in the sentence “*Our house, a brick bungalow is located at Oak Street,*” the salient entities are identified as “*house*” and “*brick bungalow*.” From this example, both entities are not identified as “*instance*” and are referred as inanimate concepts by the Generic Relation Extraction component (105), therefore a triple with the generic relation “*hasType*” is identified between the subject “*house*” and object “*brick bungalow*”. Hence, a triple of <*house*, *hasType*, *brick bungalow*> is added into the knowledge base (111).

15

On the other hand, if a sentence is identified as a non-appositive phrase type in decision 203, the Generic Relation Extraction component (105) retrieves all salient entities that are identified as in step 213. These entities which are managed by a vector list, are then determined if these entities all already processed as in decision 214. If all these entities are already processed, these entities are checked if these entities are defined in the Linguistic Resource (109) or Linked Data (108) as in decision 217 shown in FIG. 2 (b).

20

However, if there are still some entities that have yet to be processed as in decision 214, each entity in the list is selected and checked if the entities exist in the knowledge base (111) as in step 214 and decision 216. If an entity is not in the knowledge base (111), the entity is then checked if it is defined in the Linguistic Resource (109) or Linked Data (108) as in decision 217 shown in FIG. 2 (b).

25

If it is found in the Linguistic Resource (109), all extraction patterns are iterated and a pattern entry is selected by the Generic Relation Extraction component (105) as in decision 218 to check if the pattern can be applied, wherein each pattern has a corresponding property type. If a pattern can be applied, the Generic Relation Extraction component (105) adds the triple of the <*subject*, *property*, *object*> to the knowledge base (111) as in step 223 and step 227a shown in FIG. 2 (c).

30

35

For example, when a sentence is identified as non-appositive and 4 salient entities are retrieved from the sentence, wherein the entities are *Entity 1*, *Entity 2*, *Entity 3* and *Entity 4*, and one of the entities which is *Entity 4* is not found in the knowledge base, the Generic Relation Extraction component (105) then checks if *Entity 4* is identified in the Linguistic Resource (109). If *Entity 4* is identified in the Linguistic Resource (109), the Generic Relation Extraction component (105) retrieves the description. If *Entity 4* has the concept “*basement*” with a description of “*the lowest portion of a structure partly or wholly below ground level; often used for storage*”, the Generic Relation Extraction component (105) then iterates through all the extraction patterns. An example of a selected pattern entry is shown below:

Extraction-pattern#22:

15	Pattern	Property
	-----	-----
	“... <i>basement of a building</i> ...”	<i>part-of</i>

wherein from the selected pattern entry, the Generic Relation Extraction component (105) adds a triple of <*basement*, *part-of*, *building*> into the knowledge base (111) when the pattern can be applied to the description of the concept found in the Linguistic Resource (109).

However, if there is no pattern that can be applied as in decision 218, sense descriptions of the salient entity can be extracted from the Linguistic Resource (109) and transformed into context semantic structures as in step 219a. These context semantic structures, which are the conceptual structures or the graph representations, are then stored into a context index, which is an index that stores all possible descriptions, with the following information which is <*entity*, *context semantic structure*, *hypernym*> as in step 220a. In selecting the best candidate, which is a context index with the best sense description, the Resolution component (107) retrieves all context semantic structures from the context index and determines if all the entries are already processed as in decision 222. If all context semantic structures from the context index are already processed in decision 222, the process ends. If not, the Resolution component (107) performs graph similarity matching

between the identified sentence and its context semantic structures to finally add the best candidate triple information into the knowledge base (111) as in steps 224 to 227b.

5 For example in a sentence, *“Joshua Bell who is a superstar violinist has recently taken up conducting and is now a music director of the Academy of St. Martin in the Fields,”* the Generic Relation Extraction component (105) refers to the context index to get 2 sense descriptions which are *<“Joshua Bell”, dbpedia-owl:abstract, “Joshua David Bell (born December 9, 1967) is an American Grammy Award-winning violinist”>* and *<“Joshua Bell”, dbpedia-owl:abstract, “Joshua L. Bell (born November 14, 1986) is an American professional baseball player who is a free agent. A third baseman, Base has played in Major League Baseball for the Baltimore Orioles and Arizona Diamondbacks”>*, wherein these 2 sense descriptions can then be transformed into context semantic structures.

15

On the other hand, if the entity is not found in the Linguistic Resource (109) as in decision 217, the Semantic Relation Discovery component (106) then checks in the Linked Data (108) and extracts an abstract description that defines the entity as in step 219b. Similar to step 220a, the abstract description extracted from the Linked
20 Data (108) is then transformed into context semantic structure and is stored into the context index with the following information which is *<entity, context semantic structure, type property>* as in step 220b and step 221.

For instance, in the same example given previously which is *“Joshua Bell who
25 is a superstar violinist has recently taken up conducting and is now a music director of the Academy of St. Martin in the Fields,”* the information for the entity “Joshua Bell” can be retrieved from the Linked Data (108) and the information related to a violin player may be represented as *<“Joshua Bell”, dpedia-owl:occupation, “musician”>*; *<“Joshua Bell”, rdf:type, “Artist”>*; *<“Joshua Bell”, dbpedia-owl:abstract, “Joshua
30 David Bell (bornDecember 9, 1967) is an American Grammy Award-winning violinist”>* and so forth. Furthermore, information related to an athlete who plays the baseball can also be retrieved from the Linked Data (108) and may be represented as *<“Joshua Bell”, rdf:type, dbpedia-owl:BaseballPlayer>*; *<“Joshua Bell”, dbpedia-owl:abstract, “Joshua L. Bell (born November 14, 1986) is an American professional*

baseball player who is a free agent. A third baseman, Base has played in Major League Baseball for the Baltimore Orioles and Arizona Diamondbacks"> and so forth.

Based on the previous example, the Resolution component (107) distinguishes the correct sense between "baseball player" or "violinist" to extract the abstract description of the correct sense before the abstract information is being transformed into a semantic structure as in steps 224 to 227b.

A detailed explanation of steps 224 to 227b is provided herein below with respect to FIGS 3 (a – d). FIG. 3 (a) shows a whole graph structure with its sub graphs illustrated in FIGS. 3 (b - d). In FIG. 3 (b), it shows a sub graph of the sentence "*Joshua Bell who is a superstar violinist has recently taken up conducting and is now a music director of the Academy of St. Martin in the Fields.*" 2 descriptions are found for the entity "*Joshua Bell*" which are "*Joshua David Bell (born December 9, 1967) is an American Grammy Award-winning violinist*" and "*Joshua L. Bell (born November 13, 1986) is an American professional baseball player who is a free agent. A third baseman, Bell has played in Major League Baseball for the Baltimore Orioles and Arizona Diamondbacks.*" These 2 descriptions are then transformed into complete graphs.

20

One sub graph for each description is shown in FIG. 3 (b) and FIG. 3 (c) respectively, wherein FIG. 3 (b) shows a sub graph of a first sense which is "*violinist*" with a description of "*Joshua David Bell (born December 9, 1967) is an American Grammy Award-winning violinist,*" and wherein FIG. 3 (c) shows a sub graph of a second sense which is "*baseball player*" with a description of "*Joshua L. Bell (born November 13, 1986) is an American professional baseball player who is a free agent. A third baseman, Bell has played in Major League Baseball for the Baltimore Orioles and Arizona Diamondbacks.*"

From the examples given, the sentence sub graph as shown in FIG. 3 (b) and the first sense sub graph as shown in FIG. 3 (c) are said to be almost similar because the concepts "superstar-violinist" and "American-violinist" have a smaller distance in concept hierarchy if compared to the concept "baseball player," wherein a concept hierarchy refers to the taxonomy of the Linguistic Resource (109). Due to that, the first sense is said to be a better candidate than the second sense, wherein

35

all the triple information that belongs to the first sense is added into the knowledge base (111).

To choose the best candidate and to add the triple information into the knowledge base (111), the Resolution component (107) initially retrieves all context semantic structures from the context index which is represented by <subject, context semantic structure, property, object> as in step 224. Next, the Resolution component (107) performs semantic graph matching for each of the context semantic structures and returns the best match candidate triple as in step 225. To perform semantic graph matching, all sub graphs attached to the verb for each of the candidate context semantic structures are identified. All the concepts from the sentence are iterated and checked if the concept conforms to the semantic constraint for each of the concepts in the sub graph.

An example of the semantic structure for a concept “musician” is *[animate-being]<-(agnt)<-[play]->(thme)->[violin]*. For the agnt role, there are 2 semantic constraints, wherein one is on its left and one is on its right. The one on the left is an animate being and the one on the right is an action verb. Since the concept violin-player conforms to the agnt left constraint, which means the concept falls under the concept hierarchy of the animate being concept, this concept can be merged to the semantic structure to produce *[musician]<-agnt<-[play]->(thme)->[violin]*.

Once the semantic constraint is met and once the concept can be merged to the semantic structure to produce a new sub graph, a sub graph count, SG_j is added with a value of 1, wherein eventually, all context entries have their own total sub graph count. Thereon, all the sub graph counts are consolidated and all the sub graphs are merged once the iteration has been completed. Finally, a top context entry, T_c with the highest total sub graph counts, G_j is selected as in step 226, before the triple <subject, property, object> is added into the knowledge base (111) as in step 227b. For example, if context entry 1, T_1 , context entry 2, T_2 and context entry 3, T_3 have their own total sub graph count of 5, 7 and 3 respectively, T_2 , which is the top context entry will be selected before its triple of <subject, property, object> is added into the knowledge base.

While embodiments of the invention have been illustrated and described, it is not intended that these embodiments illustrate and describe all possible forms of the invention. Rather, the words used in the specifications are words of description rather than limitation and various changes may be made without departing from the scope
5 of the invention.

CLAIMS

1. A system (1000) for generating knowledge base (111) automatically comprises of:

- a) a Knowledge Base Generator (100);
- b) a Linked Data (LD) (108);
- c) a Linguistic Resources (LR) (109); and
- d) an Extraction Patterns component (110) to support a series of process and techniques for the Knowledge Base Generator (100),

wherein the system (1000) is characterised in that the Knowledge Base Generator (100) includes:

- i. a Concept Pruning component (101) to identify salient entities or the right concepts to be included;
- ii. a Concept Taxonomy Derivation component (102) to identify how a particular concept relates to its parent concept by referring to the Linked Data (108) or the Linguistic Resource (109);
- iii. a Property Taxonomy Derivation component (103) to arrange all derived relations into a hierarchy;
- iv. an Individuals Identification component (104) to detect individual and its concept type that is involved in a sentence;
- v. a Generic Relation Extraction component (105) to extract generic relations which can be discovered from the Linguistic Resource (109) or the Extraction Patterns (110);
- vi. a Semantic Relation Discovery component (106) to discover implicit relations that the Generic Relation Extraction component (105) is not able to discover; and
- vii. a Resolution component (107) to resolve a best relation candidate that can be applied to a context of a text as there are many possible relation candidates.

2. A method for generating knowledge base (111) automatically is characterised by the steps of:

- a) tokenising a paragraph into sentences;
- b) identifying if a sentence has an appositive phrase type by a Generic Relation Extraction component (105);
- c) identifying if the appositive phrases are essential or non-essential;

- d) referring to a vector list to determine the current position for each of the appositive phrases;
 - e) retrieving the entity prior to the appositive phrase;
 - f) identifying all salient entities from sentences to extract generic relations;
 - g) performing a check on all the pairs; and
 - h) adding possible pairs which represent the relations of a particular concept to another concept and to a knowledge base (111).
3. The method as claimed in claim 2, wherein the step of performing a check on all the pairs includes:
- a) determining if either one of the entities or the prior entity of the pairs is a name;
 - b) identifying the entity as an "*instance*" if the entity is a name; and
 - c) adding the "*instance*" into an individual index.
4. The method as claimed in claim 3, wherein if the entity is not identified as an "*instance*", the steps include:
- a) determining if either one of the entities or the prior entity is an animate; and
 - b) identifying the entity with a generic relation of "*is-a*" between subject and object of the entity if the entity is an animate.
5. The method as claimed in claim 4, wherein if either one of the entities or the prior entity is not an animate the entity is identified with a generic relation of "*hasType*" between subject and object of the entity.
6. The method as claimed in claim 2, wherein if a sentence is identified as a non-appositive phrase type, the steps include:
- a) retrieving all salient entities that are identified by the Generic Relation Extraction component (105), wherein these entities are managed by a vector list;
 - b) selecting each entry in the vector list and checking if the entities exist in a knowledge base (111);

- c) checking the entity if it is defined in a Linguistic Resource (109) or a Linked Data (108) if the entity is not in the knowledge base (111);
 - d) selecting a pattern entry if the pattern can be applied, wherein each pattern has a corresponding property type; and
 - 5 e) adding a triple to the knowledge base (111) if the pattern can be applied.
7. The method as claimed in claim 6, wherein if there is no pattern can be applied, the steps include:
- 10 a) extracting sense descriptions of the salient entity from the Linguistic Resource (109);
 - b) transforming the sense description into context semantic structures;
 - c) retrieving all context semantic structures from context index in selecting the best candidate by a Resolution component (107);
 - 15 d) performing graph similarity matching between identified sentence and the context semantic structures; and
 - e) choosing and adding the best candidate triple information into the knowledge base (111).
- 20 8. The method as claimed in claim 7, wherein the step of choosing and adding the best candidate triple information includes:
- a) retrieving all context semantic structures from the context index by the Resolution component (107);
 - b) performing semantic graph matching for each of the context semantic
 - 25 structures;
 - c) identifying all sub graphs attached to the verb for each of the candidate context semantic structures;
 - d) checking if the concept conforms to the semantic constraint for each of the concepts;
 - 30 e) merging the concept to the semantic structure once the concept conforms to the semantic constraint to produce a new sub graph;
 - f) adding a sub graph count with a value of 1;
 - g) consolidating all the sub graph counts;
 - h) merging all the sub graphs once the iteration is complete;

- i) selecting a top context entry with the highest total sub graph counts;
and
- j) adding the best candidate triple information into the knowledge base.

- 5 9. The method as claimed in claim 6, wherein if the entity is found in the Linked
Data (**109**), the steps include:
- a) extracting an abstract description that defines the entity; and
 - b) transforming the abstract description into context semantic structure.

10

15

20

25

30

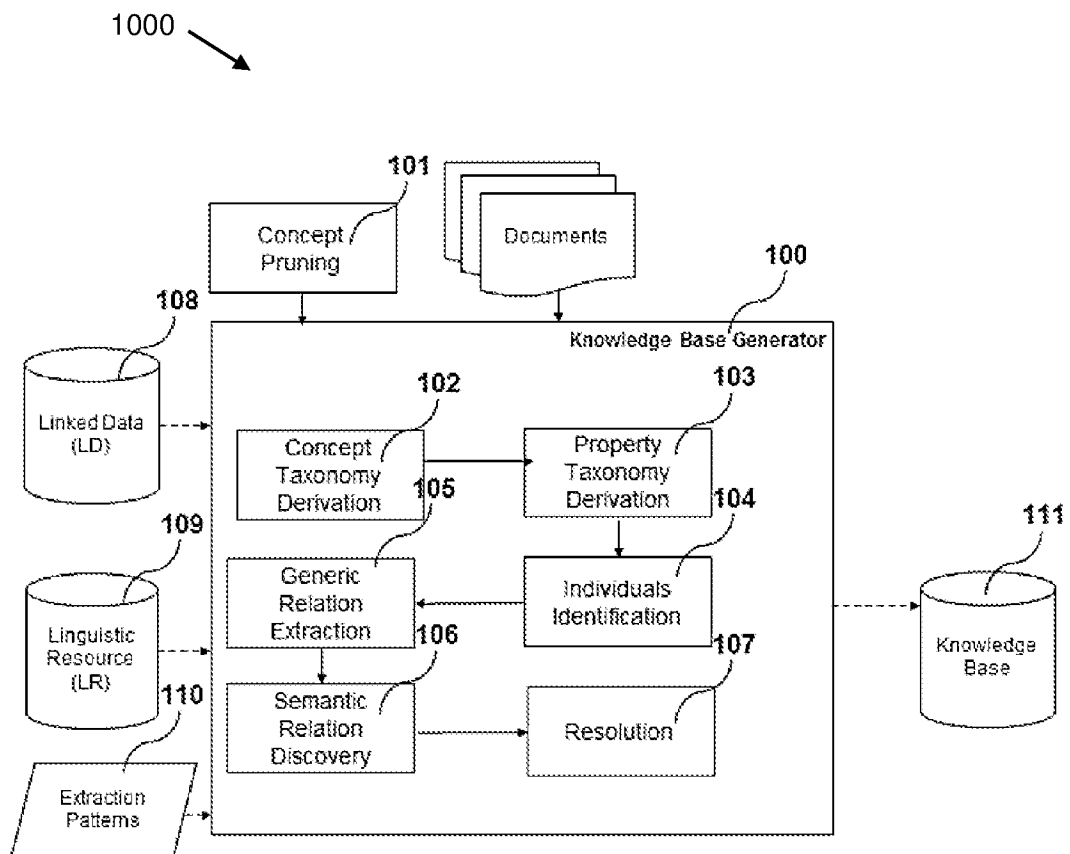


FIG. 1

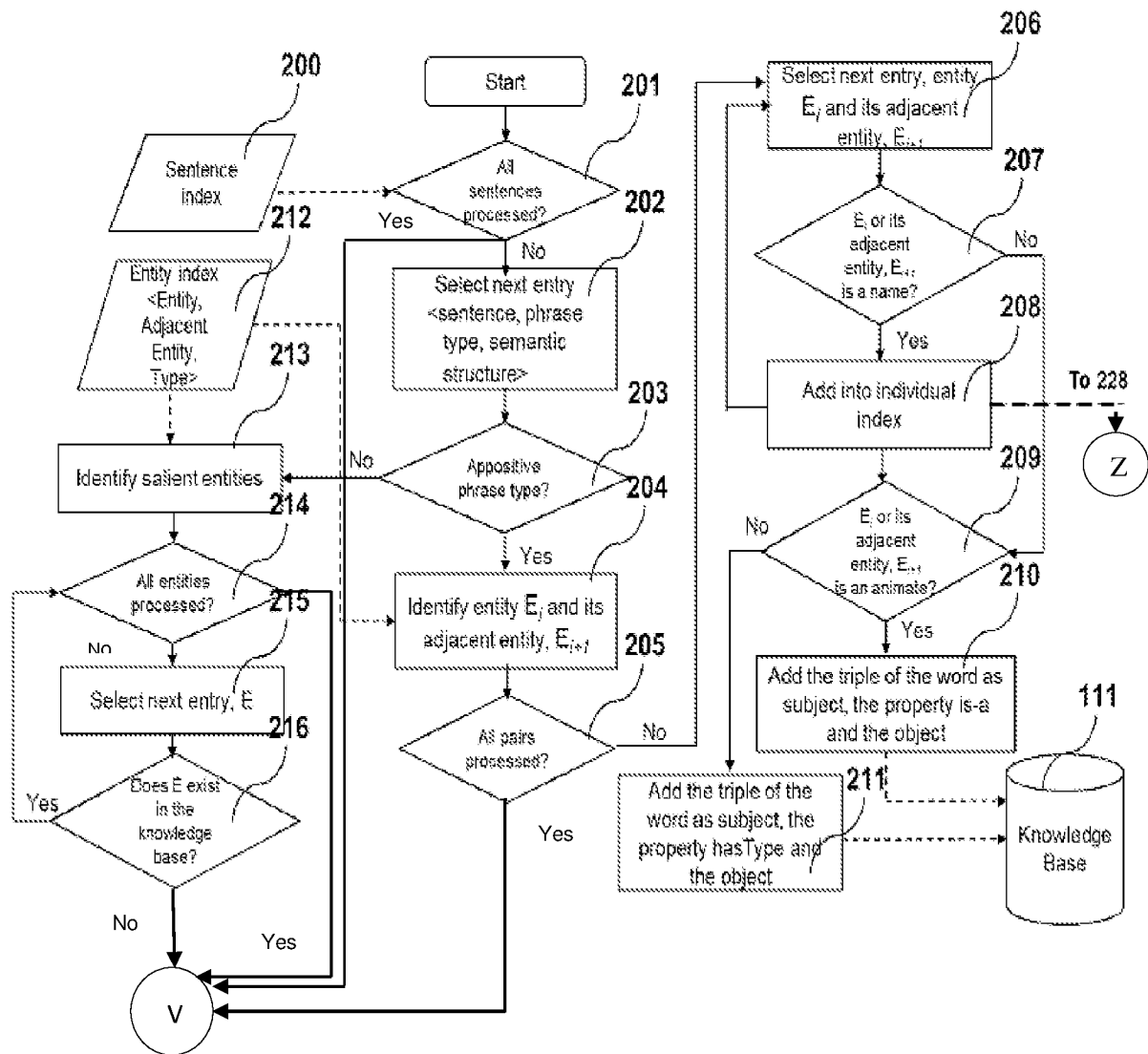


FIG. 2 (a)

3/5

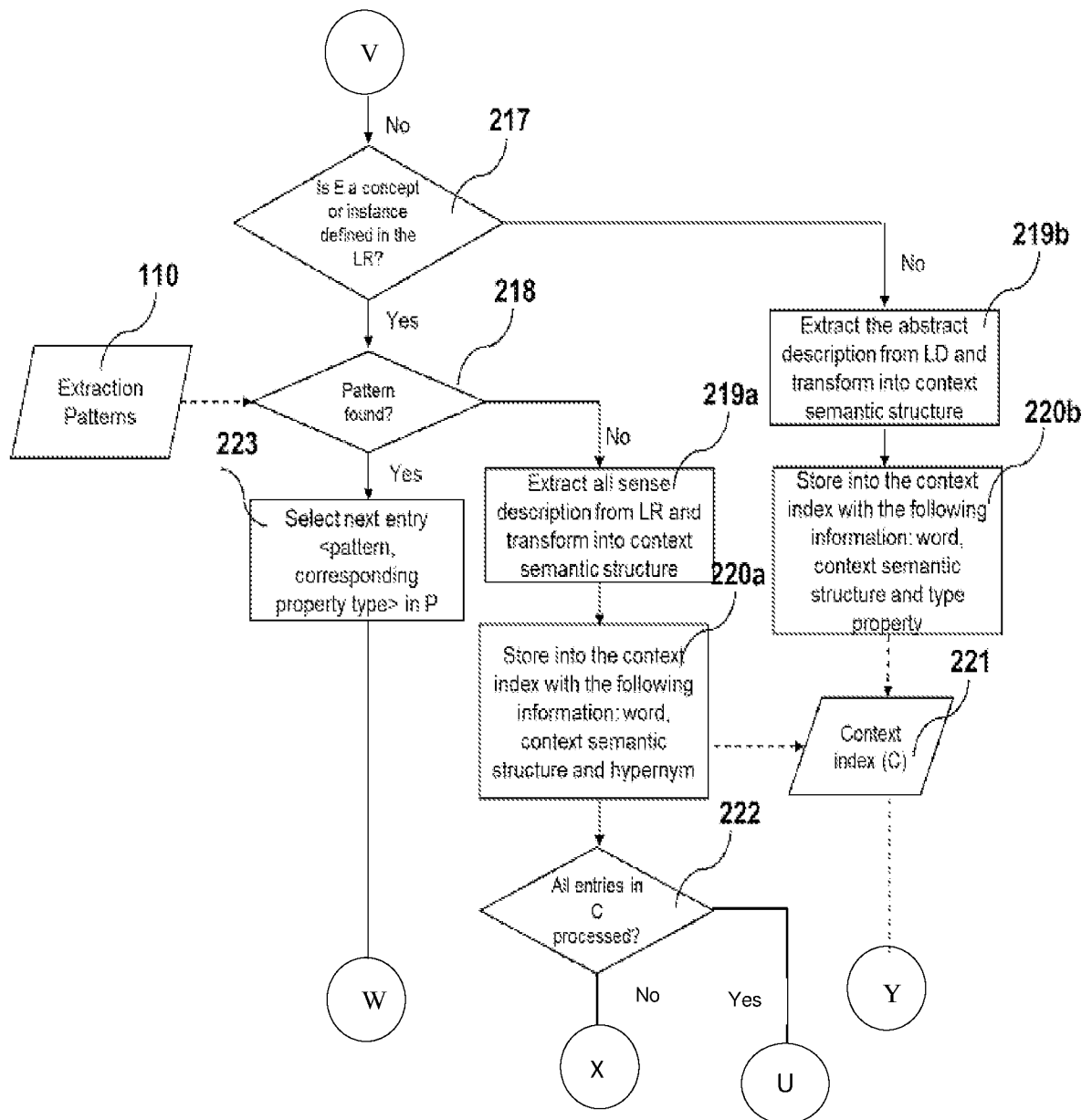


FIG. 2 (b)

4/5

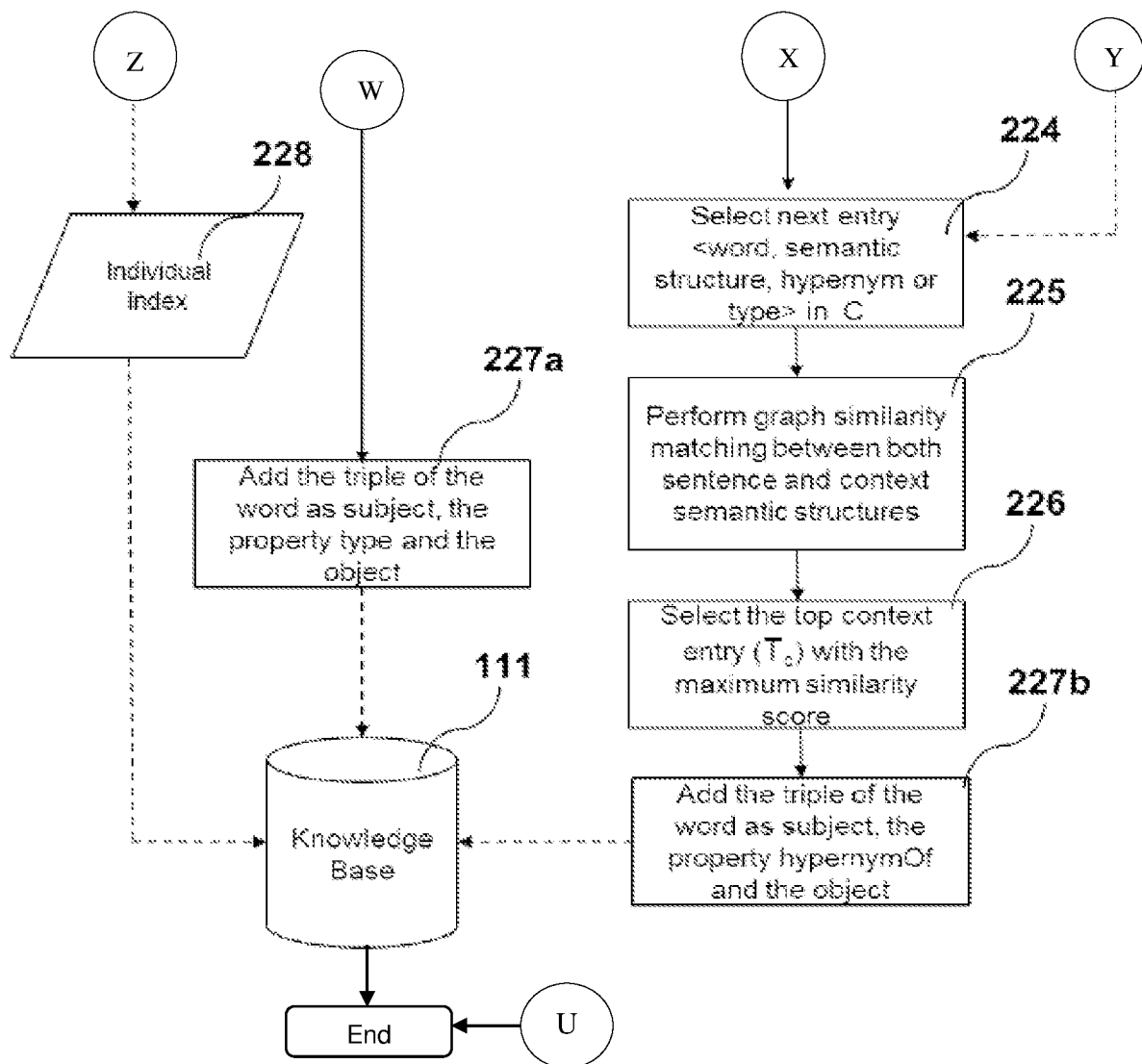


FIG. 2 (c)

5/5

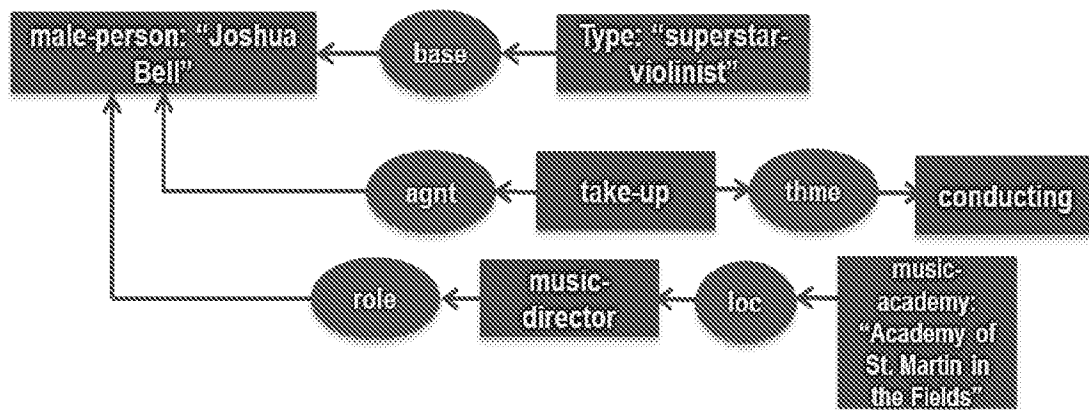


FIG.3 (a)



FIG. 3 (b)



FIG. 3 (c)



FIG. 3 (d)

INTERNATIONAL SEARCH REPORT

International application No
PCT/MY2015/050025

A. CLASSIFICATION OF SUBJECT MATTER
INV. G06F17/27
ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 6 076 088 A (PAIK WOOJIN [US] ET AL) 13 June 2000 (2000-06-13)	1-6
Y	abstract; figures 1-5 Sections 1.0-4.1	7-9
Y	----- WO 02/063493 A1 (2028 INC [US]; DAMLE ADITYA) 15 August 2002 (2002-08-15) abstract Sections III. - V1.	7-9
X	----- SERGEI NIRENBURG ET AL: "Learning by Reading by Learning to Read", SEMANTIC COMPUTING, 2007. ICSC 2007. INTERNATIONAL CONFERENCE ON, IEEE, PISCATAWAY, NJ, USA, 1 September 2007 (2007-09-01), pages 694-701, XP031138706, ISBN: 978-0-7695-2997-4 figure 1 -----	1



Further documents are listed in the continuation of Box C.



See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search

7 October 2015

Date of mailing of the international search report

14/10/2015

Name and mailing address of the ISA/

European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040,
Fax: (+31-70) 340-3016

Authorized officer

Bowler, Alyssa

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No

PCT/MY2015/050025

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 6076088 A	13-06-2000	US 6076088 A	13-06-2000
		US 6263335 B1	17-07-2001

WO 02063493 A1	15-08-2002	US 2004093328 A1	13-05-2004
		WO 02063493 A1	15-08-2002
