Semantic Web 1 (2016) 1–5
IOS Press

C2K: Acquiring Knowledge from Categories Using Semantic Associations

Editor(s): Name Surname, University, Country
Solicited review(s): Name Surname, University, Country
Open review(s): Name Surname, University, Country

Ji-Seong Kim [a,*], Dong-Ho Choi [b] and Key-Sun Choi [b]
[a] *Department of Computer Science, KAIST, 291, Daehak-ro, Yuseong-gu, Daejeon, Republic of Korea*
E-mail: jiseong@kaist.ac.kr
[b] *Department of Computer Science, KAIST, 291, Daehak-ro, Yuseong-gu, Daejeon, Republic of Korea*
E-mail: zmal0103@kaist.ac.kr, kschoi@kaist.ac.kr

Abstract. There are several RDF (Resource Description Framework) knowledge bases that store community-generated categories of entities and conceptual or factual information about entities. These two types of information may have strong associations; for example, entities categorized in *People from Korea* (categorial information) have a high probability of being a person (conceptual information) and being born in Korea (factual information). This kind of associations can be used for extracting new conceptual or factual information about entities. In this paper, we propose a prediction system that predicts new conceptual or factual information from categories of entities. First, the proposed system uses a novel association rule mining (ARM) approach that effectively mines rules encoding associations between categories of entities and conceptual or factual information about entities contained in existing RDF knowledge bases. Our extensive experiments show that our novel ARM approach outperforms the state-of-the-art ARM approach in terms of the prediction quality and coverage of these kind of associations. Second, the proposed system ranks and groups the mined rules based on their predictability by our novel semantic confidence measure calculated with a semantic resource such as WordNet. The experiments show that our novel confidence measure outperforms the standard confidence measure frequently used in the traditional ARM field in terms of discriminating the predictability of mined rules. Last, the proposed prediction system selects only rules of predictability from ranked and grouped rules, and then uses them to predict accurate new information from categories of entities. The experiments show that the results of the proposed prediction system are fairly comparable to that of the state-of-the-art prediction system in terms of the accuracy of prediction while overwhelming the coverage of prediction.

Keywords: Information Extraction, Knowledge Base Enrichment, Wikipedia Category, Semantic Association

1. Introduction

Recently, there have been a lot of attentions on information extraction from structured and unstructured data to populate existing RDF knowledge bases to expand linked open data (LOD). Many researches have been focused on text [17,13,9,16] and table [1,3,12,8] as a source of information to populate existing knowledge bases, however, categories have got relatively less attention despite of their rich information can be leveraged. This paper mainly focus on information extraction from a category that comprises both category name (unstructured part) and category hierarchy (structured part) to populate conceptual and factual information about entities in existing RDF knowledge bases.

There have been several RDF knowledge bases (KBs) that store both community-generated categories of entities and conceptual or factual information about entities in the form of a RDF triple; for example,

*Corresponding author. E-mail: jiseong@kaist.ac.kr

DBpedia [1,3,12,8] contains RDF triples encoding Wikipedia categories, concepts, factoids about entities, which are called category triple, concept triple, fact triple respectively; we call concept triple and fact triple together as a knowledge triple. In the most of the cases, there are strong associations among these types of triples; for example, entities categorized in *People from Korea* (categorial information) are usually a person (conceptual information) and have a high probability of being born in Korea (factual information). This kind of associations can be leveraged for predicting new knowledge triples from category triples.

Associations between category triples and knowledge triples can be discovered by existing association rule mining (ARM) systems such as AMIE+ [4]; however, existing approaches largely depend on only the frequency of triples as a feature to discover the associations; for example, if a KB has the sufficient number of entities that are categorized in *People from Korea* and are born in Korea in common, the rule

$$\langle x, \textit{categorizedIn, People from Korea}\rangle \Rightarrow \langle x, \textit{birthPlace, Korea}\rangle$$

is mined. However, this kind of rules also can be mined and further plentifully discovered by leveraging lexical patterns in category names; for example, we can mine the rule if lexical pattern *People from* in category *People from Korea* implies the *birthPlace* relation. This paper proposes a novel ARM approach that mainly utilizes lexical and hierarchical information of categories as a feature to mine association rules, and show that these features are more effective in discovering associations between category triples and knowledge triples in KBs than the frequency of triples.

Because mined rules encode any possible associations, to select only the predictive rules useful in the prediction of new knowledge, we need to rank and group them based on their predictability (i.e., the precision of their prediction). The standard confidence measure used in the traditional ARM field can be used and is effective to some degrees, however, it is the fact that the standard measure only uses the frequency of triples as a feature in deriving confidence values, which causes assigning high confidence values to not predictive rules; for example, the rule

$$\langle x, \textit{categorizedIn, People born in Korea}\rangle \Rightarrow \langle x, \textit{deathPlace, Korea}\rangle$$

has a high confidence value because people born in Korea tend to also die in Korea; this inappropriate situation can be resolved by semantic information contained in category name; for example, a lexical *born* in the category name is an antonym of *death* in the name of the relation. In this paper, we propose a novel semantic confidence measure that reflects not only the frequency of triples, but also the semantic distance between category name and name of relations to effectively discriminate the predictability of discovered associations.

More precisely, our contributions are as follows:

1. A novel ARM approach that mainly uses the lexical and hierarchical information of categories as a feature to effectively mine rules that encode the associations between category triples and knowledge triples.
2. A novel semantic confidence measure that is effective for discriminating the predictability of mined rules.
3. The accurate huge amount of new knowledge not captured in existing KBs, which are predicted by the predictive rules that are discovered by the combination of our novel ARM approach, semantic confidence measure, and predictive rule selection scheme.

In Chapters 2 and 3, we describe the preliminaries and defines the problem of this paper formally. In Chapter 4 of this paper, we explore the existing state-of-the-art approaches that handle the same or similar issues. In Chapter 5, we describe the proposed approach in considerable detail. In Chapter 6, we apply the proposed approach to the existing KBs and analyze the results in detail. In the last chapter, we make a conclusion.

2. Preliminaries

2.1. Categories of Entities

An entity is something that has an existence actually or potentially, concretely or abstractly, physically or not.

Entities are grouped in an **entity category** according to particular characteristics they share; e.g., birthplace or date, joint owner, thematic relevance, and so on.

One of the most frequently used naming conventions for an entity category is to express their name as a noun phrase with some special characters such as

hyphens(-) and commas(,); e.g., in Wikipedia, the large portion of category names are expressed in a noun phrase with some special characters as shown in Figure 1.

1989 births, **2008** deaths
→ (**birthYear**) births, (**deathYear**) deaths
Alumni of **King's College, Cambridge**
→ Alumni of (**education**)
Philosophers who committed **suicide**
→ (**occupation**) who committed (**causeOfDeath**)
People of **Rhode Island** in the American Civil War
→ (**type**) of (**residence**) in the American Civil War

Fig. 1. Examples of category names and possibly implied facts (bold-faced) about the entities

In most cases, a noun phrase-expressed category name delivers conceptual or factual information about an entity as shown in Figure 1.

A category also can be categorised into other categories based on various relations between them; e.g. subsumption relation, thematic relatedness, and so on. We call the set of all relations between categories as a **category hierarchy** which can be expressed as a labeled directed graph which of node denotes entity or category, edge denotes a relation between two entities, and label denotes a name of relation.

This paper focuses on a noun phrase-expressed category name and category hierarchy to extract conceptual or factual information about an entity.

2.2. Category Triples and Knowledge Triples

In this paper, we mainly focus on manipulating entities and their categories defined in an RDF (Resource Description Framework) KB. An RDF KB is a set of RDF triples that take the form of $\langle$*Subject, Predicate, Object*$\rangle$ which encodes an entity at Subject has some relation expressed in Predicate with an entity or literal value at Object. An RDF triple can be used for encoding categorial, conceptual, or factual relations between entities; e.g., $\langle$*Tim Berners-Lee, categorizedIn, English computer scientists*$\rangle$ encodes a categorial relation between the entity and the category, $\langle$*Tim Berners-Lee, type, person*$\rangle$ and $\langle$*Tim Berners-Lee, birthplace, London*$\rangle$ encodes a conceptual and factual relation between the two entities; We call these triples as **category triple**, **concept triple**, and **fact triple** respectively; We call both concept triple and fact triple together as a **knowledge triple**.

2.3. Association Rules and C2K Rules

An **association rule** encodes an association among triples with some shared variables; for example, the rule

$$\langle x, \textit{residence, England}\rangle, \langle x, \textit{deathPlace, England}\rangle \Rightarrow \langle x, \textit{birthPlace, England}\rangle$$

encodes that if entity x both lived in England and died in England, x tends to be born in England. An association rule can be used for encoding an association between category triple and knowledge triple; for example, the rule

$$\langle x, \textit{categorizedIn, 1912 births}\rangle \Rightarrow \langle x, \textit{birthYear, 1912}\rangle$$

encodes that if entity x is categorized in category *1912 births*, x tends to born in *1912*. We refer to this special case of an association rule as a **C2K rule** which comprises one category triple on its left side and one conceptual or factual triple on its right side. In this paper, we mainly focus on mining C2K rules to predict conceptual or factual information from category triples.

3. Problem Statement: C2K Rule Mining

Let E be a set of entities; R, a set of relations; L, a set of literals such as integers or floating point numbers; and C, a set of entity categories.

A set of n instances of category relations can be represented as a set of category triples, $T_{cat} = \{\langle e_i, categorizedIn, c_i\rangle\}_{i=1}^{n}$ where $e \in E$, $categorizedIn \in R$, $c \in C$.

A set of m instances of conceptual or factual relations can be represented as a set of knowledge triples, $T_{know} = \{\langle e_i, r_i, o_i\rangle\}_{i=1}^{m}$, where $e \in E$, $r \in R - \{categorizedIn\}$, and $o \in (E \cup L)$.

The problems to be dealt with can be described as follows.

The first is to mine rules of the form, $t_{cat}^{x} \Rightarrow t_{know}^{x}$ where $t_{cat}^{x} = \langle x, categorizedIn, c \in C\rangle$ and $t_{know}^{x} = \langle x, r \in R - \{categorizedIn\}, o \in E \cup L\rangle$ where x denotes a variable for an entity.

The second is to rank the mined rules in order of their predictability (i.e., the precision of their prediction)

The last is to select only predictive rules from the ranked rules.

The overall goal is to predict new knowledge triples (i.e., conceptual or factual information about entities) from the given T_{cat} and T_{know} in existing KBs by C2K rules of the predictability, which is discovered by solutions of the above-mentioned three problems.

4. Related Work

4.1. The Rich Source of Knowledge: Categories

In 2007, Auer et al. initiated the DBpedia project that originally extracted information from Wikipedia infoboxes and encapsulated it in RDF triples [1,3,12, 8]. The project successfully extracted 18M triples, after being further developed. However, the fact that only about 44.2% articles have infoboxes, which results in only a minor portion of the articles being covered by these infobox-driven triples, while the categories cover about 80.6% articles to extract triples [10]. It implies that categories are an important and rich source of knowledge, which is worth an in-depth study.

4.2. Heuristic Rule-Based Approach

In 2007, Suchanek et al. [15] presented YAGO1 that is an RDF KB constructed from Wikipedia infoboxes and categories. The triples in YAGO1 mainly comprise two parts, infobox-driven triples and category-driven triples. They extract category-driven triples by applying manually crafted rules to Wikipedia categories; for example, if an entity x is categorized in a category c, and c shows the lexical pattern *Cities in y*, then a triple $\langle x, \textit{locatedIn}, y\rangle$ is extracted. This rule can be rewritten into multiple C2K rules such as follows.

$$\langle x, \textit{categorizedIn, Cities in Germany}\rangle, \\ \Rightarrow \langle x, \textit{locatedIn, Germany}\rangle$$

$$\langle x, \textit{categorizedIn, Cities in England}\rangle, \\ \Rightarrow \langle x, \textit{locatedIn, England}\rangle$$

...

In 2014, Mahdisoltani et al. [11] presented YAGO3 that is an expansion of YAGO1 with triples of spatio-temporal and multilingual information; however, it seems that the extraction approach for category-driven triples has not the significant difference to that of YAGO1 according to their report [11].

As their approach solely depends on a few rules curated by limited human efforts, only a few categories are leveraged as the source of extraction and only a few relations defined in YAGO are covered by the rules; This limitation inspires us to design a more automatic approach that can learn C2K rules by itself with minimal human-intervention to achieve high coverage of extraction while keeping the high precision of results.

4.3. Association Rule-Based Approach

In 2015, Galárraga et al. [4] proposed AMIE+ that is an association rule mining (ARM) system that automatically mines association rules among triples in KBs. They reported that AMIE+ overwhelmed the former ARM systems, WARMR [5] and ALEPH[1], in terms of the predictability of mined rules and the scalability over the size of KBs. AMIE+ is the most promising model to solve our problem, C2K rule mining, however, she is premature to be directly applied to our problem; AMIE+ only uses the frequency of triples as a feature to mine association rules. Because category names contain usable lexical and semantic information, the existing ARM system can be further optimized for the C2K rule mining problem by leveraging such additional features. Our approach uses not only frequency of triples, but also lexical and semantic information of category and entity names to enhance the existing ARM approaches to be more suitable for the C2K rule mining problem.

5. Proposed Approach

Figure 2 shows the overall workflow of the proposed prediction system; in the first stage, the system refines an input KB by filtering out erroneous knowledge triples in order to reduce the corrupted result by error propagation; Second, the system normalizes category and knowledge triples of the input KB; Third, the system begins rule mining with the normalized triples and bootstraps mined rules further; Fourth, mined rules are ranked in order of their predictability and only those of predictability are selected by human-intervention; Lastly, new knowledge triples are predicted from existing category triples by selected predictive rules. In the following sections, we detail each process.

[1] http://www.cs.ox.ac.uk/activities/machlearn/Aleph/aleph_toc.html

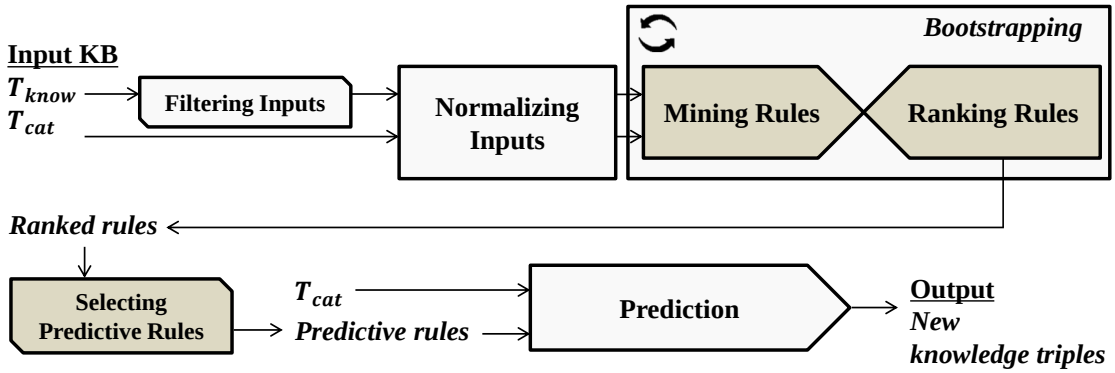

Fig. 2. The workflow of the proposed prediction system

5.1. Filtering Erroneous Input Triples

Because automatically constructed KBs such as DBpedia mostly contain erroneous triples, we can mine erroneous association rules between category triples and erroneous knowledge triples. To be worse is that if we bootstrap mined rules, the size of errors can be hugely populated by error propagation; it is a main weakness of distantly supervised learning. As the association rules of errors are not of our interest, we simply filter out errors in input KBs to keep the quality of results as follows.

1. **Measuring object-type frequencies of each relation:** From knowledge triples in KBs, we measure object-type frequencies of each relation. Table 1 shows an example of the object-type frequencies of the *birthPlace* and *genre* relations in DBpedia 2015 (the most general types such as *thing* are excluded). The table shows that the two relations have entities of the inappropriate types (*album*, *cricket team*, ...) as their object.
2. **Filtering out erroneous knowledge triples**: We filter out knowledge triples whose object has a type frequency that is lower than an average object-type frequency of a relation encoded in the knowledge triples; for example, in Table 1, knowledge triples that encode the *birthPlace* relation and have *album* and *cricket team*-typed entities as their object are filtered out.

dbo:birthPlace		*dbo:genre*	
Object-type	Frequency	Object-type	Frequency
country	313,150	music genre	328,772
settlement	230,641	string	108,414
city	224,341	genre	11251
...	...	...	...
album	4,450	**cricket team**	881
film	1,009	**language**	139
species	473	**film**	131
...	...	...	...
Average	**7,945**	Average	**4,782**

Table 1

Object-type frequencies of *birthPlace* and *genre* relations in DBpedia 2015

5.2. Normalizing Input Triples

For effectiveness of the C2K rule mining, the proposed system normalizes input category triples and filtered knowledge triples as follows:

- Word Segmentation: Several categories, entities, and relations defined in KBs sometimes have a name comprises just one word that is a concatenation of several words (e.g., causeOfDeath). The ARM approach presented in this paper uses lexical-based string matching in mining rules. To maximize the lexical matching for effective rule mining, the system transforms the concatenated words of names into normal word sequence and decapitalize them (e.g., causeOfDeath → cause of death).
- Word Stemming: As the purpose of the word segmentation, to maximize the lexical matching for effective rule mining, we stem all individual words of names (e.g., died in New York → die in New York).

An example of the normalization is shown in Table 2. In the rule mining process, all input triples are used as their normalized form, and then after rules are

mined, they are restored from the normalization. In the rest of this paper, we show examples of triples as the normalized form for the sake of simplicity and consistency.

5.3. C2K Rule Mining

In this section, we describe the proposed ARM approach in detail.

5.3.1. Checking Lexical Match

When mining C2K rules, the proposed system checks for two match-conditions, and then mines the C2K rules according to satisfied conditions. The match-conditions are as follows.

Exact Match-Condition: If there are $\langle e$, *categorize in,* $c\rangle \in T_{cat}$ and $\langle e, r, o\rangle \in T_{fact}$ such that c is exactly matched with o, then the system will mine a C2K rule, as follows:

$$\langle x, \textit{categorize in}, c\rangle \Rightarrow \langle x, r, c\rangle$$

where x denotes a variable for an entity $e \in E$.

For example, if there are a category triple, $\langle$*john mccarthy, categorize in, computer scientist*$\rangle$, and a fact triple, $\langle$*john mccarthy, occupation, computer scientist*$\rangle$, then the rule

$$\langle x, \textit{categorize in, computer scientist}\rangle \Rightarrow \langle x, \textit{occupation, computer scientist}\rangle$$

will be mined by satisfying the exact match-condition.

Partial Match-Condition: If there are $\langle e$, *categorize in,* $c\rangle \in T_{cat}$ and $\langle e, r, o\rangle \in T_{fact}$ such that o is a partial word sequence of c, then the system will learn a lexical pattern p_r for r, as follows:

$$p_r = c - o$$

where $c-o$ represents the remaining part of c that is not matched with o.

For example, if there are a category triple, $\langle$*artificial intelligence, categorize in, 2001 film*$\rangle$, and a knowledge triple, $\langle$*artificial intelligence, premiere year, 2001*$\rangle$, *x film* will be a lexical pattern of $p_{premiereYear}$, where x denotes a variable for an object of the *premiere year* relation.

Then, rules are mined by applying learned lexical patterns to a set of candidate categories, as following steps:

Step 1: The system gathers a set of candidate categories that will be compared with p_r. The set of candidate categories can be defined as follows:

$$C_{candi} = \{c\} \cup \textit{siblings}(c)$$

where $\textit{siblings}(x)$ denotes the set of siblings of a category x on a category hierarchy.

Step 2: If $c_{candi} \in C_{candi}$ is partially matched with a learned lexical pattern p_r, then the system will mine a C2K rule, as follows:

$$\textit{rule} : \langle x, \textit{categorize in}, c_{candi}\rangle \Rightarrow \langle x, r, c_{candi} - p_r\rangle$$

where $c_{candi} - p_r$ represents the remaining part of c_{candi} that is not matched with p_r and x denotes a variable for entity $e \in E$.

Step 3: If $c_{candi} - p_r$ does not have the same datatype of o or does not belong to at least one of the same categories of o (i.e., it does not share the same parent categories), *rule* will be discarded.

For example, if there are a learned lexical pattern, $p_{premiereYear} = x$ *film*, and a candidate category, *2015 film*, the rule

$$\langle x, \textit{categorize in, 2015 film}\rangle \Rightarrow \langle x, \textit{premiere year, 2015}\rangle$$

will be mined by Step 2. Because *2001* and *2015* have the same datatype, *an integer*, the mined rule will not be filtered out by Step 3. Another example is that if there is a learned lexical pattern, $p_{premiereYear} = x$ *film*, and a candidate category ,*american film*, the rule

$$\langle x, \textit{categorize in, american film}\rangle \Rightarrow \langle x, \textit{premiere year, american}\rangle$$

will be mined by Step 2, but because *2001* and *american* do not have the same datatype and do not share the same parent categories, the mined rule will be discarded by Step 3.

In all of the lexical matching, the proposed system uses words of similar meaning according to WordNet to maximize possible matches; for example, if there are a category triple $\langle$*john lennon, categorize in, people from england*$\rangle$ and a knowledge triple $\langle$*john lennon, type, person*$\rangle$, *people from england* will be partially matched to *person* because *people* and *person* have similar meaning according to WordNet. By leveraging semantic information of Wordnet, the system can learn the lexical pattern, *x from england*, for the *type* relation despite of no lexical matches.

	Unnormalized	Normalized
S	Alan Turing	alan turing
P	categorizedIn	categorize in
O	Mathematicians who committed suicide	mathematician who commit suicide

Table 2
An example of normalization

5.3.2. Bootstrapping Mined Rules

A KB is usually unbalanced with regard to the number of category triples and knowledge triples; for example, some entities have many category triples but few knowledge triples (possibly not at all); this situation causes much possible matches not to be occurred, which results in the scarcity of associations discovered. It would be overcome by using a predicted result (i.e., new knowledge triples) as a new part of input knowledge triples. For this reason, the proposed system bootstraps input KB and mined rules with several iterations. In every iteration, the system bootstraps the KBs with new knowledge triples predicted by the previously discovered predictive rules that have the confidence value of more than θ_c, and then, repeatedly bootstraps mined rules through the remaining iterations. If the ratio of the increase in the mined rules falls below some threshold θ_q, the bootstrapping is stopped, and the final output is the lastly mined rules with their own confidence values.

5.4. Ranking and Grouping C2K Rules

The confidence of rules can be used for measuring rules' prediction quality. If it is possible to assign high confidence values to predictive rules and low ones to the others, we are able to rank and group them according to their predictability; this is a crucial part to guarantee high quality prediction. In this section, we introduce the standard confidence measure frequently used in the traditional ARM field, then, our specially designed confidence measure for more effective ranking and grouping of C2K rules than the standard measure.

5.4.1. Standard Confidence Measure

Before defining the standard confidence, we should define a support that is a basic ingredient of the standard confidence measure. A support of a set of triples is defined as the number of entities for x, which occupy a subject of a given set of triples $\{\langle x, r_i, o_i\rangle\}_{i=1}^{n}$, as follows:

$$supp(\{\langle x, r_i, o_i\rangle\}_{i=1}^{n}) = \sum_{e \in E} \mathbb{I}(\{\langle e, r_i, o_i\rangle\}_{i=1}^{n})$$

A support delivers how large evidences in a given data supports an association rules.

The standard confidence of an association rule is a probability that entities occupying a subject of a given set of category triples $\{t_{cat}^x\}$ also occupy a subject of a given set of knowledge triples $\{t_{know}^x\}$, as follows:

$$Conf_{Stand}(t_{cat}^x \Rightarrow t_{fact}^x) = \frac{supp(\{t_{cat}^x, t_{fact}^x\})}{supp(\{t_{cat}^x\})}$$

where $t_{cat}^x = \langle x, categorizedIn, c \in C\rangle$ and $t_{know}^x = \langle x, r \in R - \{categorizedIn\}, o \in E \cup L\rangle$.

A standard confidence delivers the prediction quality of a rule. It can be used to rank and group C2K rules, however, it only uses the occurrence of triples in a KB as a feature despite of the fact that there are a lot of other features such as lexical and semantic information in category names, entity names, and relation names. In the following sections, we introduce novel lexical and semantic confidence measures that mirror such diverse information.

5.4.2. Proposed Confidence Measures

Lexically Averaged Confidence: The standard confidence is not capable of grouping C2K rules with regard to their lexical similarity; for example, in Figure 5.2, $rule_1$ and $rule_2$ have the different standard confidence values, i.e., they are not grouped although category names of them ($t_{cat_1}^x$ and $t_{cat_2}^x$) share the same lexical pattern (*people from x*). Because the same lexical pattern usually delivers the same relational information, it is desirable to group C2K rules according to their shared lexical units. In this regard, we define the **lexically averaged (LA) confidence** measure and also some parameters to be tolerant to the sparsity of KBs, as follows:

$$Conf_{LA}(rule) = \frac{[\sum_{(t_{cat}^x \Rightarrow t_{fact}^x) \in SP(rule)} supp(\{t_{cat}^x, t_{fact}^x\})] \wedge \theta_u}{[\sum_{(t_{cat}^x \Rightarrow t_{fact}^x) \in SP(rule)} supp(\{t_{cat}^x\})] \vee \theta_l \wedge \theta_u}$$

where $SP(rule)$ indicates a set of all rules mined by the same lexical pattern that is used for mining *rule*, θ_l and θ_u represent the lower and upper bound parame-

ters, respectively, and $\wedge$ and $\vee$ denote the min and max functions, respectively.

A lower bound parameter (θ_l) of the LA confidence measure prevents an abnormally high confidence values (e.g., $standConf(rule_1)$ in Figure 3), which is caused by sparse category triples (e.g., $supp(t^x_{cat_1})$). An upper bound parameter (θ_u) prevents an abnormally low confidence values, which is caused by sparse knowledge triples. The LA confidence values for $rule_1$ and $rule_2$ shown in Figure 3 are calculated as follows ($\theta_l = 10$ and $\theta_u = 100$).

$$Conf_{LA}(rule_1) = Conf_{LA}(rule_2) = 0.1$$

As shown in the right above result, LA confidence is capable of grouping C2K rules with regard to the lexical similarity in contrast with the standard confidence shown in Figure 3.

Semantically Adjusted LA Confidence: In the previous, we rank and group C2K rules by the frequency of triples and lexical similarities. In addition to the frequency and lexical information, there is another useful feature, semantic information of category names and relation names; for example, the category *settlement establish in 1870* and the relation *founding year* have a highly related meaning on the basis of contained meanings of words (*establish* and *founding*). With semantic information, C2K rules can be further ranked and grouped by their semantic plausibility. In this regard, we define **semantically adjusted LA (SALA) confidence** that is a compound of the semantic distance between category triple and knowledge triple with LA confidence of a rule. The SALA confidence uses synonym, member-holonym, and member-meronym classes in WordNet to calculate the semantic similarity between category name and relation name. The more category name and relation name contain words of the aforesaid three classes in common, the more they are considered to have the similar meaning. The semantic dissimilarity between category name and relation name is also calculated by using antonym classes in the same way as the similarity.

In the following, we define the SALA confidence step by step. In the first step, we define the similarity distance between two names as a cosine distance between the vector representation of the two names, as follows.

$$cos(n_1, n_2) = \frac{vec(n_1) \cdot vec(n_2)}{\|vec(n_1)\|\|vec(n_2)\|}$$

where $vec(x)$ denotes the vector representation of x

In the second step, we define two functions that calculate the similarity distance and the dissimilarity distance, respectively, between category name c and relation name r, as follows.

$$sim(c,r) = \frac{\sum_{w_r \in W(r)} \vee(cos(w_s \in S(w_r), w_c \in W(c)))}{|W(r)|}$$

$$dSim(c,r) = \frac{\sum_{w_r \in W(r)} \vee(cos(w_d \in D(w_r), w_c \in W(c)))}{|W(r)|}$$

where $W(x)$ denotes the set of the space-split words of x; $S(x)$ represents the set of the synonyms, member-holonyms, and member-meronyms of x; $D(x)$ indicates the set of the antonyms of x; and $\vee$ refers a max function.

The $sim(c,r)$ and $dSim(c,r)$ functions indicates how many words in a relation name (i.e., $w_r \in W(r)$) are semantically covered by words in a category name (i.e., $w_c \in W(c)$). Finally, we define the (unnormalized) SALA confidence that ranges from -1.0 to 2.0, as follows.

$$Conf_{unnormSALA}(rule) = Conf_{LA}(rule) + sim(c,r)^{w_{sim}} - dSim(c,r)^{w_{dSim}}$$

where c denotes the name of a category contained in *rule*, r represents the name of relation encoded in a knowledge triple contained in *rule*, and w_{sim} and w_{dSim} denote a weight value.

The normalized SALA confidence that ranges from 0.0 to 1.0 is defined as follows.

$$Conf_{normSALA}(rule) = (Conf_{unnormSALA}(rule) + 1)/3$$

In our experiments, we calculate SALA confidence of mined rules by weakening the similarity score via w_{sim} and strengthening the dissimilarity score via w_{dSim}; the reason is that the occurrence of dissimilar words tends to be more influential in meaning than the occurrence of similar words; for example, given category name *19th-century deaths* and relation name *birth year*, it is clear that the two names have opposite meaning despite the fact that they have the same number of similar words (*century* in the category name and *year* in the relation name) and dissimilar words (*death* in the category name and *birth* in the relation name).

$$t^x_{cat_1} = \langle x, \textit{categorized in, people from korea} \rangle$$
$$t^x_{cat_2} = \langle x, \textit{categorized in, people from japan} \rangle$$
$$t^x_{know_1} = \langle x, \textit{birthplace, korea} \rangle$$
$$t^x_{know_2} = \langle x, \textit{birthplace, japan} \rangle$$
$$supp(\{t^x_{cat_1}, t^x_{know_1}\}) = 1, supp(\{t^x_{cat_1}\}) = 1$$
$$supp(\{t^x_{cat_2}, t^x_{know_2}\}) = 1, supp(\{t^x_{cat_2}\}) = 10$$
$$rule_1 = t^x_{cat_1} \Rightarrow t^x_{know_1}$$
$$rule_2 = t^x_{cat_2} \Rightarrow t^x_{know_2}$$
$$Conf_{Stand}(rule_1) = \frac{supp(\{t^x_{cat_1}, t^x_{know_1}\})}{supp(\{t^x_{cat_1}\})} = 1.0$$
$$Conf_{Stand}(rule_2) = \frac{supp(\{t^x_{cat_2}, t^x_{know_2}\})}{supp(\{t^x_{cat_2}\})} = 0.1$$

Fig. 3. An example of different standard confidence values for $rule_1$ and $rule_2$ mined by the same lexical pattern, *people from x*

With the SALA confidence measure, we can rank and group C2K rules by not only the occurence of triples, but also lexical and semantic similarities. The proposed prediction system uses SALA confidence as its main confidence measure to rank and group mined C2K rules. In the experimental sections, we show the SALA confidence is more effective in discriminating the predictability of C2K rules than the standard confidence.

5.5. Selecting Predictive C2K Rules

In this section, we describe the approach for selecting only predictive rules from all mined C2K rules by minimal human-intervention.

After rule mining is done, mined rules are ranked and grouped according to their own SALA confidence values; if rules are mined by the same lexical pattern, they will be grouped into the same lexical group; lexical groups are, then, reorganized into a semantic group according to the semantic plausibility of them. For lexically and semantically grouped rules, there can be two options for only selecting predictive ones from them, as follows.

- **Selecting by a confidence threshold:** We can simply select predictive rules whose confidence value is higher than a predetermined threshold. The advantage of this approach is that predictive rules can be fully automatically selected without any human-intervention. However, there is a danger of selecting not predictive rules because of the fact that a confidence value is hard to be perfectly matched to the actual precision of rules.
- **Selecting by the manually estimated precision of grouped predictions:** The alternative way for selecting predictive rules is to select rules by the manually estimated precision of their predictions. However it is exhaustive when there are a huge number of rules to be estimated. To minimize human effort, the proposed system uses a group-based sampling approach as follows.

1. **Group** rules according to a relation they can predict, i.e., a relation encoded in the right-hand side of a rule.
2. **Divide** a group of rules into five subgroups according to their confidence values (e.g., the first subgroup have rules with confidence values of more than 0.8, the second group have rules with confidence values of less than 0.8 and more than 0.6, and so on).
3. **Predict** triples using the rules of each subgroup, and then estimate the precision of predicted triples subgroup-by-subgroup by sampling and evaluating their predictions.
4. **Select** only the rules, whose subgroup's precision is estimated larger than some threshold, as predictive rules.

This approach needs far less human effort than the exhaustive approach because of well grouped rules with the SALA confidence; for example, our one researcher spent about just 30 minutes for selecting 22,538 predictive rules of averagely 90% precision from 28,150 mined rules through this approach; if he had executed exhaustive selection, 30 minutes would have been too short to estimate the precision of the entire 28,150 rules. In the experimental sections, we will show that the huge amount of new knowledge triples that are pre-

dicted by selected rules of this approach show a fairly high precision.

6. Evaluation and Experiments

This chapter shows the comparison of mining, ranking, and prediction capabilities between the proposed approaches and the existing other approaches.

6.1. Common Experimental Settings

All of the experiments in this paper share some common experimental settings: The parameters of the proposed ARM approach, θ_c and θ_q, are set as 0.9 and 0.1 respectively. The parameters of the proposed confidence measure, θ_l and θ_u, are set as 5 and 1000 respectively. We use English WordNet 3.0 to calculate semantic similarity and dissimilarity for deriving a SALA confidence value. The weight values, w_{sim} and w_{dSim}, for the SALA confidence are set by 2 and 1/2 respectively.

6.2. Comparison of Mining Capability: The Proposed ARM Approach vs. AMIE+

6.2.1. Experimental Settings

In this experiment, we compare the proposed ARM approach to the state-of-the-art ARM system, AMIE+, in terms of prediction quality and coverage of mined C2K rules. We use the DBpedia 3.2 dataset as an input KB; we use the category triples of DBpedia 3.2 as input category triples and randomly sampled 30% of the mapping-based properties and types of DBpedia 3.2 as input knowledge triples. The remaining 70% of the mapping-based properties and types of DBpedia 3.2 are used as an answer that will be compared to the prediction of C2K rules mined by both the approaches. AMIE+ is configured to have only one category triple as the body of a rule and a knowledge triple as the head of a rule. AMIE+ is enabled to mine rules with a constant value. Other parameters of AMIE+ are used as the default values set in the released software[2].

[2] https://www.mpi-inf.mpg.de/departments/databases-and-information-systems/research/yago-naga/amie/

6.2.2. Evaluation of The Mined Rules

We mine C2K rules with both the approaches, and then predict new knowledge triples by mined C2K rules. We measure the prediction qualities of both the approaches by measuring a hit ratio that indicates how large predictions are matched with answers, which delivers the prediction quality of rules. The hit ratio is defined as follows.

$$hitRatio = \frac{hitCount}{predictionSize}$$

where *hitCount* denotes the number of predicted triples that exactly matched with the answer triples and *predictionSize* represents the number of predicted triples.

Table 3 shows the results obtained by both the approaches. The table shows that the hit ratio of the proposed approach is about two times larger than that of AMIE+, which means that the prediction quality of the C2K rules mined by the proposed approach is about two times better than that of AMIE+. It is also noticeable that the proposed approach overwhelms AMIE+ in terms of the coverage of categories (*catCov*) and entities (*entCov*), which means the C2K rules mined by the proposed approach can leverage more categories for knowledge acquisition and enrich a larger variety of entities with new knowledge than those mined by AMIE+. The table also shows the spending time of both the approaches. Although the proposed approach uses simple memoization and indexing techniques to reduce the overall mining time, AMIE+ is overwhelmingly faster than the proposed approach mainly because of the following two reasons:

- The proposed approach inherently depends on a string matching function that incurs a heavy cost in mining.
- AMIE+ is highly optimized in terms of the efficiency on speed by effective pruning operations, which is one of AMIE's main contributions.

Although the proposed approach needs much time for overall rule mining than AMIE+, it is still reasonably fast to enrich KBs because it spends only few hours on the entire DBpedia dataset, not days or months.

In this experiment, although the proposed approach is better than AMIE+ in terms of the prediction quality and coverage of mined C2K rules, it does not indicates the proposed approach is totally better than AMIE+ in terms of the whole rule mining problem; AMIE+ is capable of mining more general forms of rules than C2K

	Approach	*hitRatio*	*catCov*	*entCov*	Time
P	Ours	0.06	0.4	0.70	28.52m
	AMIE+	0.03	0.004	0.28	1.7m
T	Ours	0.4	0.09	0.26	90.76s
	AMIE+	0.23	$3.02 \times e^{-6}$	0.12	26.9s

Table 3

Comparison between the prediction qualities of both the approaches: the prediction quality is delivered by *hitRatio*, the coverage of categories (*catCov*), and the coverage of entities (*entCov*). P and T mean that mapping-based properties (P) or types (T) of DBpedia is used as input knowledge triples

rules. What we would like to argue in this experiment is that the lexical, hierarchical, and other available information of some entities such as categories is more useful in mining predictive rules than just using the occurrence of triples in KBs.

6.3. Comparison of Ranking Capability: SALA Confidence vs. Standard Confidence

6.3.1. Experimental Settings

In this experiment, we compare the proposed SALA confidence to the standard confidence in terms of a ranking capability. Some C2K rules are manually evaluated as a predictive or not predictive rule to be used for the comparison. To fairly compare the two measures, we use normalized SALA confidence that ranges from 0.0 to 1.0 which is the same as the standard confidence.

6.3.2. Evaluation Methods

To compare the two measures, we define the relative discriminating power (RDP) of the SALA confidence against the standard confidence as follows:

$$RDP_{SALA}(rule) = \begin{cases} Conf_{SALA}(rule) - Conf_{Stand}(rule), \\ \text{if } rule \text{ is predictive} \\ Conf_{Stand}(rule) - Conf_{SALA}(rule). \\ \text{if } rule \text{ is not predictive} \end{cases}$$

The idea is that a confidence measure assigning a higher confidence value to a predictive rule is more discriminating than the other measure. A positive RDP_{SALA} score means that the SALA confidence is more discriminating than the standard confidence while a negative RDP score delivers the opposite meaning.

We group mined C2K rules into five groups according to their confidence values (e.g., the first group has confidence values of more than 0.8, the second group has confidence values of less than 0.8 and more than 0.6, and so on). RDP_{SALA} scores for each group is calculated by averaging the confidence values of all rules contained in a group. When an averaged RDP_{SALA} score is close to 1.0, it means the SALA confidence is more close to an actual precision than the standard confidence. On the contrary, when an averaged RDP_{SALA} score is close to -1.0, it means the standard confidence is more close to an actual precision than the SALA confidence. With the averaged RDP_{SALA} score, we can verify which of the two measures is more appropriate for discriminating the predictability of mined C2K rules.

6.3.3. Evaluation Results

Figure 4 shows the result of the comparison. Almost every histogram shows that the SALA confidence is more discriminating than the standard confidence. In Figure 5 and 6, examples of the outstanding discriminating power of the SALA confidence against the standard confidence are denoted with arrows. To conclude, the last histogram in Figure 4 tells us that on average, the SALA confidence is 0.22 points more close to an actual precision than the standard confidence.

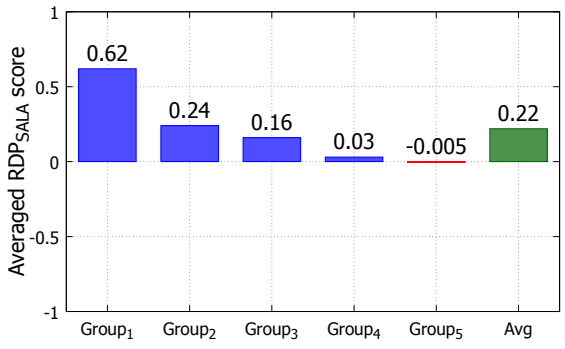

Fig. 4. Averaged RDP_{SALA} scores: When an averaged RDP_{SALA} score is close to 1.0, it means the SALA confidence is more close to an actual precision than the standard confidence. The opposite is true when an averaged RDP_{SALA} score is close to -1.0.

6.4. Comparison of Prediction Capability: The Proposed System vs. YAGO

6.4.1. Experimental Settings

In this experiment, we compare the proposed prediction system with both YAGO1's and YAGO3's prediction systems. Although all the versions (YAGO1, YAGO2, YAGO3) of YAGO's prediction systems seem to use the similar C2K approach, which are not of significant difference according to their reports [6,7,2,11], we use YAGO1 (the initial version) and YAGO3 (the latest version) for the fair comparison. DBpedia 3.2 dataset is used as an input KB for the comparison with YAGO1 and DBpedia 2014 is used as an input KB for the comparison with YAGO3, because the Wikipedia dump dates corresponding to these DBpedia datasets are the almost same as those of the YAGO1 and YAGO3 datasets. We use the category triples of the DBpedia dataset as input category triples and use all of the infobox properties, mapping-based properties, and mapping-based types of the DBpedia dataset as input knowledge triples for the proposed prediction system.

6.4.2. Evaluation Methods

We manually evaluate the precision of the triples predicted by the proposed system, and then compare the evaluated results to that of YAGO1, relation-by-relation. In the case of the comparison with YAGO3, because YAGO3 evaluated their results by a Wilson center, we also use the Wilson center for estimate the accuracy of our results. Because DBpedia's properties and YAGO's properties have different names, we only compare those whose meanings are the almost exactly same. In the manual evaluation, a predicted triple is regarded as a true positive when the source category of the triple implies the predicted triple; Wikipedia articles also be referenced to verify that the information encoded in the triple is essentially true.

6.4.3. Evaluation Results

Table 4 and 5 show the comparison between the results of the proposed system and YAGO1's and YAGO3's prediction system. The triples predicted by the proposed system are totally extracted from Wikipedia categories (denoted by C in the table) while the triples predicted by YAGO's prediction system are extracted from not only Wikipedia categories, but also Wikipedia infoboxes (denoted by I in the table). All relations in the table are shown based on DBpedia's properties, however, it should be noted that we only compare YAGO's properties whose meaning is the almost exactly same as DBpedia's properties. The table shows that in some relations (the top-line bold-faced relations: origin, birthPlace, genre, homeTown, writer, party, award, birthYear, and deathYear), the proposed system is more predictive than YAGO's prediction system while in some relations (the bottome-line bold-faced relations: deathPlace, prize, director, foundationYear, and type), YAGO's prediction system is more predictive than the proposed system. The results show that both the systems have their own specialized region. The main difference between the two systems is the coverage of relations. The proposed system extracts entirely about 4,450 relations from only the categories, while YAGO's prediction system extracts entirely about 100 relations from the categories and infoboxes [14]. Although the entire results of the proposed system should be evaluated, however, it is clear that the proposed system can predict more diverse relations among entities than YAGO's prediction system. This high coverage is mainly achieved by the automatic lexical pattern learning of the proposed ARM approach; the proposed ARM approach discovers associations between category triples and knowledge triples by automatically learning lexical patterns of each pre-defined relation of DBpedia; whereas, YAGO's prediction system discovers the associations by the human-specified lexical patterns of relations. In conclude, the proposed prediction system can predict new knowledge with the high coverage of relations by the automatic lexical pattern learning while maintaining the high precision comparable to the human capability.

6.5. Prediction Examples

Figures 5 and 6 show examples of the true positives and false positives of the proposed system's prediction. The figures show the predicted triples per relation, the standard and unnormalized SALA confidence values of the rules used to predict the triples, and the source categories of the triples. Figure 5 shows the knowledge triples successfully predicted from various categories with the fairly improved SALA confidence values (denoted with arrows). Figure 6 shows the knowledge triples that should not be predicted; these triples have the almost zero or negative SALA confidence values (denoted with arrows).

6.6. C2K on Datasets of Different Languages

To see that the proposed approaches can be applied on other language datasets, we apply the proposed sys-

	Ours			YAGO1		
Relation	Amount	Precision	Source	Amount	Precision	Source
dbo:origin	439,472	0.97	C	11,497	0.97	I
dbo:birthPlace	234,161	1.0	C	36,189	0.96	I
dbo:genre	222,519	0.94	C	106,797	0.94	I
dbo:homeTown	189,368	0.98	C	11,497	0.97	I
dbo:writer	49,207	0.92	C	12,469	0.96	I
dbo:party	45,140	0.96	C	6,198	0.97	C
dbo:nationality	731,561	0.81	C	4,865	0.96	I
dbo:place	558,245	0.83	C	60,261	0.97	C
dbo:dateOfBirth	539,053	0.8	C	441,274	0.96	C
dbo:birthYear	428,181	1.0	C	441,274	0.96	C
dbo:country	134,426	0.74	C	4,865	0.96	I
dbo:establish	115,810	0.95	C	110,930	0.97	C
dbo:location	57,712	1.0	C	60,261	0.97	C
dbo:deathPlace	3,511	1.0	C	13,618	0.96	I
dbo:prize	10,217	1.0	C	23,076	0.96	C
dbo:director	13,316	0.99	C	23,723	0.96	I
dbo:foundationYear	63,618	0.95	C	110,830	0.96	C
rdf:type	764,848	0.96	C	4,505,603	0.97	C

Table 4

Evaluation of the proposed system (using the English DBpedia 3.2 dataset) and YAGO1's prediction system: C and I in Source columns mean the results are predicted from the Wikipedia categories and Wikipedia infoboxes, respectively.

	Ours		YAGO3	
Relation	Amount	Accuracy	Amount	Accuracy
dbo:award	134,184	0.97	55,935	0.96
dbo:birthYear	975,218	0.97	487,521	0.96
dbo:deathYear	450,862	0.97	280,078	0.96
dbo:location	976,409	0.88	528,768	0.96
dbo:foundingYear	175,172	0.96	574,071	0.94
dbo:type	4,449,460	0.96	10,622,967	0.97

Table 5

Evaluation of the proposed system (using the English DBpedia 2014 dataset) and YAGO3's prediction system: All relations are predicted only from Wikipedia categories. The accuracy of our results is evaluated by a Wilson center with averagely 4.65 width.

tem on the Korean DBpedia 2014 dataset. To normalize the dataset, we implement Korean stemmer and word segmenter. To calculate SALA confidence values, we extract Korean synonym and antonym sets by translating and filtering synonyms and antonyms defined in English WordNet 3.0. The proposed system predicts frequently used top 21 relations in the Korean DBpedia 2014 dataset. Table 6 shows the results of the prediction. The table shows that the proposed system is fairly effective on the dataset of Korean language with regard to both the amount and precision. The result implies the possibility that the proposed prediction system can be successfully applied on many different language datasets with language-specific stemmer, word segmenter, and dictionaries.

7. Beyond the Current KBs

To verify how the large portion of a KB is enriched by the prediction of the proposed system, we measure the number of entities covered by each relation before the prediction and after the prediction, which is shown in Table 7. The table shows that relations about time (birthYear, deathYear, and foundingYear) have fairly good improvements in the coverage of entities. Relations about location (regionServed, city, country, location, and so on) and person (education, country, award, birthPlace, and so on) show also good improvements. Overall, we have enriched the existing DBpedia KB with plentiful knowledge about person, time, location,

Source Category	S	P	O	Stand. Conf.	SALA Conf.
1949 births	Kevin Volans	*birthYear*	1949	0.28 →	**0.88**
Year of birth 27 May 1956 (living people)	William Gainey	*birthYear*	1956	1.0	0.60
Mountains of Israel	Mount Arbel	*location*	Israel	0.04 →	**0.9**
Registered Historic Places in Colorado	Glenisle	*location*	Colorado	0.03 →	**1.5**
UK Unionist Party politicians	Roger Hutchinson	*party*	UK Unionist Party	0.1 →	**1.0**
Leaders of the Communist Party of China	Li Zhaoxing	*party*	Communist Party of China	0.41 →	**0.71**
Airports in Oregon	Faust Field	*type*	Airport	0.99	0.97
Oil companies of Russia	Invest Union	*type*	Company	0.32	0.32
Films directed by Bob Clampett	Tin Pan Alley Cats	*director*	Bob Clampett	0.28 →	**1.17**
Eli Roth films	Grindhouse (film)	*director*	Eli Roth	0.8	0.44
Screenplays by William Nicholson	Firelight	*writer*	William Nicholson	0.4 →	**0.92**
Songs written by Neil Thrasher	I Melt	*writer*	Neil Thrasher	1.0	0.72
Novels by Frederick Forsyth	The Deceiver (novel)	*writer*	Frederick Forsyth	0.0 →	**0.54**

Fig. 5. Examples of the true positives. Arrows(→) indicates that the unnormalized SALA confidence assigns high confidence values to predictive rules that are undervalued by the standard confidence.

and so on from categories with the high precision using the proposed prediction system.

Using the proposed prediction system, KBs with category triples can be enriched every time new category triples come in. Table 8 shows the persistently growing size of category triples in each version of DBpedia. Each column of $|T_{cat}|$ indicates the number of entire category triples in each version of DBpedia. Each column of New $|T_{cat}|$ indicates the number of category triples that do not exist in the immediately previous version of DBpedia (Only New $|T_{cat}|$ of DBpedia 3.8 is the number of category triples not contained in DBpedia 2.0). With semantic associations discovered by the proposed prediction system, we can enrich KBs beyond the current states from the persistently growing category triples.

8. Conclusion

In this paper, we have proposed a prediction system that can predict new knowledge triples from category triples in existing KBs. The proposed system mines association rules by the proposed ARM approach, ranks and groups mined rules based on their predictability with the proposed semantic confidence measure, and then predicts new knowledge triples from category triples by predictive rules selected by the grouped sampling-based selection approach. Extensive experiments have shown that the proposed ARM approach outperforms the state-of-the-art ARM system, AMIE+, in terms of the prediction quality and the coverage of mined C2K rules, which indicates that the lexical and hierarchical information of categories are more useful in mining plentiful and plausible C2K rules than the occurrence of triples. The experiments also have shown that the proposed semantic confidence measure outperforms the standard confidence measure in terms of discriminating the predictability

Source Category	S	P	O	Stand Conf.	SALA Conf.
1526 deaths	Andrea Ferrucci	*birthYear*	1526	0.02 →	**-0.52**
1919 births	Andrew Boyle	*deathYear*	1919	0.0 →	**-0.54**
Japanese public universities	Kyushu Dental College	*location*	Japanese	0.0	0.0
Leaders of the Pakistan Movement	Khawaja Nazimuddin	*party*	Pakistan Movement	0.04	0.71
American anarchists	John Perry Barlow	*party*	Anarchist	0.0	0.001
Software companies of Canada	LiveHive Systems	*type*	Software	0.013	0.013
American Film producers	Pierce Rafferty	*type*	Film	0.001	0.001
American architects	Eric J. Hill	*birthPlace*	American	0.0	0.004
Infectious disease deaths in the Ottoman Empire	Adam Mickiewicz	*birthPlace*	Ottoman Empire	0.0 →	**-0.54**
Presidents of the Cambridge Union Society	Leon Brittan	*deathPlace*	Cambridge	0.01 →	**-0.33**
David Gates albums	First (David Gates album)	*director*	David Gates	0.0	0.01
Boysetsfire albums	Before the Eulogy	*writer*	Boysetsfire	0.0	0.1

Fig. 6. Examples of the false positives. Arrows(→) indicates that the unnormalized SALA confidence assigns low confidence values to not predictive rules that are overvalued by the standard confidence.

of mined C2K rules. By our proposed system, plentiful and accurate new knowledge triples are extracted from Wikipedia categories, which is fairly comparable to the results of the state-of-the-art prediction system, YAGO, while overwhelming the coverage of relations. We showed that the proposed system can be expanded to be used in the datasets of different languages if some language-specific resources are provided. With the proposed prediction system, we can enrich KBs of various languages beyond the current states with knowledge triples of the high precision, which are predicted from persistently growing category triples.

References

[1] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives. *Dbpedia: A nucleus for a web of open data*. Springer, 2007.

[2] J. Biega, E. Kuzey, and F. M. Suchanek. Inside yago2s: A transparent information extraction architecture. In *Proceedings of the 22nd international conference on World Wide Web companion*, pages 325–328. International World Wide Web Conferences Steering Committee, 2013.

[3] C. Bizer, J. Lehmann, G. Kobilarov, S. Auer, C. Becker, R. Cyganiak, and S. Hellmann. Dbpedia-a crystallization point for the web of data. *Web Semantics: science, services and agents on the world wide web*, 7(3):154–165, 2009.

[4] L. Galárraga, C. Teflioudi, K. Hose, and F. M. Suchanek. Fast rule mining in ontological knowledge bases with amie+. *The VLDB Journal*, 24(6):707–730, 2015.

[5] B. Goethals and J. Van den Bussche. Relational association rules: Getting w armer. *Pattern Detection and Discovery*, pages 145–159, 2002.

[6] J. Hoffart, F. M. Suchanek, K. Berberich, E. Lewis-Kelham, G. De Melo, and G. Weikum. Yago2: exploring and querying world knowledge in time, space, context, and many languages. In *Proceedings of the 20th international conference companion on World wide web*, pages 229–232. ACM, 2011.

[7] J. Hoffart, F. M. Suchanek, K. Berberich, and G. Weikum. Yago2: A spatially and temporally enhanced knowledge base from wikipedia. In *Proceedings of the Twenty-Third international joint conference on Artificial Intelligence*, pages 3161–3165. AAAI Press, 2013.

Relation	Predicted Triple	New Triple	Precision
rdf:type	109,242	58,427	0.99
dbo:country	69,447	65,397	0.89
dbo:birthPlace	68,340	55,052	0.86
dbo:birthYear	46,572	46,554	1.0
dbo:genre	39,720	36,518	0.91
dbo:nationality	25,031	23,397	0.95
dbo:location	18,677	17,110	0.99
dbo:team	14,634	11,981	1.0
dbo:foundingYear	13,909	13,808	1.0
dbo:education	13,079	13,024	0.73
dbo:position	11,271	7,364	1.0
dbo:language	10,423	5,964	1.0
dbo:club	9,770	3,368	1.0
dbo:deathYear	8,906	8,896	1.0
dbo:city	8,670	8,052	0.93
dbo:regionServed	5,099	4,955	0.98
dbo:almaMater	4,911	4,903	1.0
dbo:channel	3,338	2,456	0.89
dbo:award	2,414	2,184	1.0
dbo:musicalArtist	2,210	1,480	1.0
dbo:computingMedia	1,363	1,345	1.0

Table 6

Statistics of the prediction with the Korean DBpedia 2014 dataset

Relation	Before Covered Entity	After Covered Entity	Increase Rate
dbo:birthYear	34	46,539	1,369
dbo:deathYear	25	8,918	357
dbo:foundingYear	647	13,599	210
dbo:almaMater	26	4,416	170
dbo:regionServed	284	4,759	17
dbo:city	719	7,144	10
dbo:education	2,381	12,275	5.16
dbo:country	12,542	56,993	4.54
dbo:location	4,303	17,516	4.07
dbo:award	930	2,690	2.89
dbo:nationality	11,010	28,549	2.59
dbo:genre	11,338	28,599	2.52
dbo:computingMedia	470	1,060	2.26
dbo:musicalArtist	1,385	2,729	1.97
dbo:birthPlace	23,285	44,934	1.93
dbo:team	4,678	7,558	1.62
dbo:language	6,494	9,847	1.52
rdf:type	89,256	127,421	1.43
dbo:channel	3,450	4,558	1.32
dbo:club	4,109	4,684	1.14
dbo:position	7,332	8,171	1.11

Table 7

The increasing number of entities covered by the properties defined in the Korean DBpedia 2014 dataset

DBpedia	3.8	3.9	2014	2015
$\|T_{cat}\|$	15,112,372	16,598,682	18,731,754	20,232,713
New $\|T_{cat}\|$	12,580,437	2,693,767	3,513,358	2,675,892

Table 8

The increasing number of category triples in each version of DBpedia

[8] J. Lehmann, R. Isele, M. Jakob, A. Jentzsch, D. Kontokostas, P. N. Mendes, S. Hellmann, M. Morsey, P. van Kleef, S. Auer, et al. Dbpedia-a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web Journal*, 5:1–29, 2014.

[9] M. X. C. Liu. Semantic relation classification via hierarchical recurrent neural network with attention.

[10] Q. Liu, K. Xu, L. Zhang, H. Wang, Y. Yu, and Y. Pan. Catriple: Extracting triples from wikipedia categories. In *The Semantic Web*, pages 330–344. Springer, 2008.

[11] F. Mahdisoltani, J. Biega, and F. Suchanek. Yago3: A knowledge base from multilingual wikipedias. In *7th Biennial Conference on Innovative Data Systems Research*. CIDR 2015, 2014.

[12] P. N. Mendes, M. Jakob, and C. Bizer. Dbpedia: A multilingual cross-domain knowledge base. In *LREC*, pages 1813–1817, 2012.

[13] Y. Shen and X. Huang. Attention-based convolutional neural network for semantic relation extraction.

[14] F. M. Suchanek. Automated construction and growth of a large ontology. *PhDThesis. Saarbrücken University, Germany*, 2008.

[15] F. M. Suchanek, G. Kasneci, and G. Weikum. Yago: A large ontology from wikipedia and wordnet. *Web Semantics: Science, Services and Agents on the World Wide Web*, 6(3):203–217, 2008.

[16] N. T. Vu, H. Adel, P. Gupta, and H. Schütze. Combining recurrent and convolutional neural networks for relation classification. *arXiv preprint arXiv:1605.07333*, 2016.

[17] K. Xu, Y. Feng, S. Huang, and D. Zhao. Semantic relation classification via convolutional neural networks with simple negative sampling. *arXiv preprint arXiv:1506.07650*, 2015.