Ranking Schemas by Focus: A Cognitively-Inspired Approach

Mattia Fumagalli[1], Daqian Shi[2], and Fausto Giunchiglia[2]

[1] Conceptual and Cognitive Modeling Research Group (CORE), Free University of Bozen-Bolzano, Bolzano, Italy
{mattia.fumagalli@unibz.it}
[2] Department of Information Engineering and Computer Science (DISI), University of Trento, Italy
{daqian.shi@unitn.it, fausto.giunchiglia@unitn.it}

Abstract. The main goal of this paper is to evaluate *knowledge base schemas*, modeled as a set of *entity types*, each such type being associated with a set of *properties*, according to their *focus*. We model the notion of focus as *"the state or quality of being relevant in storing and retrieving information"*. This definition of focus is adapted from the notion of *"categorization purpose"*, as first defined in cognitive psychology. In turn, this notion is formalized based on a set of knowledge metrics that, for any given focus, rank knowledge base schemas according to their quality. We apply the proposed methodology on a large data set of state-of-the-art knowledge base schemas and we show how it can be used in practice.[3]

Keywords: Knowledge base schema · Schema ranking · Categorization purpose · Knowledge representation · Mental representation

1 Introduction

Following contemporary psychology, the purpose of what we call categorization can be reduced to "*...a means of simplifying the environment, of reducing the load on memory, and of helping us to store and retrieve information efficiently*" [1,2]. According to this perspective, categorizing consists of putting things (like events, items, objects, ideas or people) into categories (e.g., classes or types) based on their similarities, or common features. Without categorization we would be overwhelmed by the huge amount of diverse information coming from the external environment and our mental life would be chaotic [3, 4]. In the context of Artificial Intelligence (AI), the purpose of categorization is usually implemented by well defined and effective information objects, namely knowledge base schemas (KBSs), where prominent examples include *knowledge graphs (KGs), schema layers* [5] and *ontologies* [6]. KBSs offer many pivotal benefits [7], such as: *i).* human understandability; *ii).* a fixed and discrete view over a stream of multiple and diverse data; *iii).* a tree or a grid structure, so that each information

[3] Data and scripts are available at https://github.com/knowdive/Focus

can be located by answering a determinate set of questions in order; and *iv).* an encoding in a formal language, which is a fragment of the first first-order predicate calculus. These benefits allow representing high-performance solutions to large-scale categorization problems, namely problems of efficient information storage and retrieval.

KBSs are the backbone of many semantic applications and play a central role in improving the efficiency of many "categorization systems" (like digital libraries or online stores). Their construction usually involves a huge effort in terms of time and domain-specific knowledge (see for instance well-known problems as "knowledge acquisition bottleneck" [8]). So far, in order to minimize the effort in building KBSs, a huge number of search engines, catalogs, and metrics have been produced, to also facilitate their reuse [9]. As the number of available KBSs increases, the definition of approaches for facilitating their reuse becomes an even greater issue [10], also considering new areas of application, see, for instance, *Relational Learning* [11] or *Transfer Learning* [12].

The main goal of this paper is to provide a quantifiable and deterministic way to assess KBSs according to their *categorization purpose*, by means of what we call their *focus*. Here we take a KBS as a set of *entity types*, each such type being associated with a set of *properties*, and we model the notion of focus as "*the state or quality of being relevant in storing and retrieving information*". We measure focus via a set of metrics that we ground on the notion of *categorization*, as first defined in cognitive psychology [13]. We then show how focus can be used to rank: *i).* the concepts inside a KBS which are more/less informative; *ii).* the concepts across multiple KBS which are more/less informative; and *iii).* the KBSs which are more/less informative. As final step, in order to test the utility of the focus measures we show how it can be used to support engineers in measuring the relevance[4] of KBSs. That is, *a).* we verify how the KBSs ranking provided by the focus metrics reflects the ranking of the KBSs provided by a group of knowledge engineers, according to guidelines inspired by a well-known experiment in cognitive psychology; *b).* we verify how focus can help scientists in selecting better KBSs to train a classifier and address an *Entity Type Recognition (ETR)* task, as it is defined in [15].

The paper main contributions can be then summarized as follows: *i).* a cognitive psychology grounded account of the notion of focus (Section 2); *ii).* a set of metrics that apply to KBSs, their entity types, and their properties, which can be used to rank KBSs according to their focus (Section 3); *iii).* an analysis of the application of the metrics over ~ 50 state-of-the-art (SoA) data sets (Section 4). Based on these results, in the second part of the paper, the scope of Section 5 is to describe the feasibility and practical utility of the approach; Section 6 discusses the related work, while Section 7 reports the conclusions.

[4] "*Something (A) is relevant to a task (T) if it increases the likelihood of accomplishing the goal (G), which is implied by T*" [14].

2 Defining focus

Imagine that, by saying "the green book on my desk in my office", someone wants someone else to bring her that book. This will happen only if the two subjects share *a way of describing objects* into those that are offices and those that are not, those that are books and those that are non-books, desks, and non-desks. These "object descriptions" are what is meant to convey for retrieving the intended objects. The point is *to draw sharp lines around the group of objects to be described*. That is *the categorization purpose* of an object description. These object descriptions, also called types, categories, or classes, are the basis of the organization of our mental life. Meaning and communication heavily depend on this categorization [3, 4, 13, 16].

Following the contemporary descriptions by psychologists, and, in particular, the seminal work by Eleanor Rosch [1], the categorization purpose of objects descriptions or categories, can be explained according to two main dimensions, namely: *i). the maximization of the number of features that describe the members of the given category* and *ii). the minimization of the number of features shared with other categories.*

To evaluate these dimensions Eleanor Rosch introduces the central notion of *cue validity* [17]. This notion was defined as "*the conditional probability* $p(c_j|f_j)$ *that an object falls in a category* c_j *given a feature, or cue,* f_j", and then used to define the set of basic level categories, namely those categories which maximize the number of characteristics (i.e., features or attributes like "having a tail" and "being colored") shared by their members and minimize the number of characteristics shared with the members of their sibling categories. The intuition is that *basic level categories* have higher cue validity and, because of this, they are *more relevant in categorization.*

Rosch's definitions were designed for experiments where humans were asked to score objects as members of certain given categories. We adapt Rosch's original methodology to the context of KBS engineering. In our setting, each available KBS (see, for instance, *schema.org*[5] or *DBpedia*[6]) plays the role of a categorization, which is modeled as a set of *entity types* associated to a set of *properties*, whose main function is to *draw sharp lines around the types of entities it contains, so that each member in its domain falls determinately either in or out of each entity type* [15, 18]. The knowledge engineers play a role similar to the persons involved in Rosch's experiment. Each knowledge base schema provides a rich set of categorization examples. Each entity type plays the role of a category and all entity type properties play the role of features. The categorization purpose of the KBS is what we call *focus*. We then model the notion of focus as "*the state or quality of being relevant in storing and retrieving information*" and we quantify the degree of this relevance by adapting Rosch's notion of cue validity as follows:

[5] `http://schema.org/`
[6] `https://wiki.dbpedia.org/`

- we take each property to have the same "cue validity" (which we assume to be normalized to one);
- for each KBS we equally divide the property "cue validity" across the entity types the properties are associated to;
- by checking the wide-spreading of "cue validity" we quantify the relevance of the KBS and entity types in storing and retrieving information.

The "focus" can be then calculated in relation to this analysis and, in turn, it can be functionally articulated in:

- *the entity types focus*, namely, what allows to identify the entity types that are maximally informative categories, which have a *higher categorization relevance*, or, more precisely, which maximize the number of properties and minimize the number of properties shared with other categories. These entity types being, to some extent, related to what expert users consider as "core entity types" or central entity types for a given domain;
- *the KBSs focus*, namely, what allows to identify the KBSs that maximize the number of maximally informative (focused) entity types. These KBSs being described, to some extent, as *"clean"* or *"not-noisy"* and being related to what expert users classify as well-designed KBSs [7].

3 Focus metrics

Taking inspiration by the research results presented in [15, 18], we assume that a KBS can be formalized as: $K = \langle E_K, P_K, I_K \rangle$, with $E_K = \{e_1, ..., e_n\}$ being the set of *entity types* of K, $P_K = \{p_1, ..., p_n\}$ being the set of *properties* of K, and I_K being a binary relation $I_K \subseteq E_K \times P_K$ that expresses specific entity types that *are associated* with specific properties. We describe that an entity type e is *associated with* a property p when e is being in the domain of the p, in formula e $\in$ *dom(p)*. For instance, the entity type *Person* can be in the domain of properties such as *address* or *name*, while the property *address* may be associated with entities such as *Person*, or *Building*. It is worth noticing that the proposed formalization of entities and properties is different from, e.g., the encoding that can be provided by the OWL[7] representational language. The key difference can be clarified by considering our formalism very similar to what is proposed by the *Formal Concept Analysis* (FCA) methods [19]. Our commitment to this model is motivated not only on foundational considerations but also on pragmatic grounds. Once properties and entity types are formalized as described above, data can be indeed analyzed and processed with few limitations in practice.[8]

Given the above formalization, we define a main set of metrics, namely the focus of an entity type, $Focus_e$ and the focus of of a KBS, $Focus_k$.

[7] `https://www.w3.org/2001/sw/WebOnt/`

[8] See [20] for an overview of the multiple available approaches and applications.

3.1 $Focus_e$

According to the notion of *entity type focus* which is introduced in Section 2, we model *entity type focus* metric $Focus_e$ as:

$$Focus_e(e) = Cue_e(e) + \eta Cue_{er}(e) = Cue_e(e)(1 + \frac{\eta}{|prop(e)|}) \ , \ \eta > 0 \tag{1}$$

In the above function, e represents an entity type. The $Focus_e$ results from the summation of Cue_e and Cue_{er}. Cue_e represents the *cue validity* of the entity type. Cue_{er} represents a normalization of Cue_e. η represents a constraint factor to be applied over Cue_{er}. The constraint factor η is used to manipulate the weight of the metric Cue_{er}, thereby affecting the value of the metric $Focus_e$. Specifically, two parts of the function can also be combined, in which $|prop(e)|$ is the number of properties associated with the specific entity type e. Notice that, in this setting, we assume that the weight of η is equal to 1, postponing the analysis on how to derive the best constraint factors to the immediate future work.

To model Cue_e and Cue_{er} we mainly adapted the work presented in [21]. In order to calculate Cue_e, firstly, we define the *cue validity* of a property p associated with an entity type e, also called Cue_p, as:

$$Cue_p(p, e) = \frac{PoE(p, e)}{|dom(p)|} \in [0, 1] \tag{2}$$

$|dom(p)|$ presents the cardinality of entity types that are the domain of the specific property p. *PoE(p, e)* is defined as:

$$PoE(p, e) = \begin{cases} 1, if \ e \in dom(p) \\ 0, if \ e \notin dom(p) \end{cases} \tag{3}$$

$Cue_p(p, e)$ returns 0 if p is not associated with e. Otherwise returns *1/n*, where n is the number of entity types in the domain of p. In particular, Cue_p takes the maximum value 1 if p is associated with only one entity type.

Given the notion of Cue_p we provide the notion of *cue validity* of an entity type. Cue_e, is related to the sum of the *cue validity* of the properties associated with the specific entity type e and is modeled as follows:

$$Cue_e(e) = \sum_{i=1}^{|prop(e)|} Cue_p(p_i, e) \in [0, |prop(e)|] \tag{4}$$

Cue_e provides the *centrality* of an entity in a given KBS, by summing all its properties Cue_p. Cue_e refers to the maximization of the properties associated with entity type e with the members it categories.

Given the notion of Cue_e, we capture the minimization level of the number of properties shared with other entity types, inside a KBS with the notion of Cue_{er}, which we define as:

$$Cue_{er}(e) = \frac{Cue_e(e)}{|prop(e)|} \in [0, 1] \tag{5}$$

After deriving Cue_e and Cue_{er} it is possible to calculate $Focus_e$. Notice that, to normalize the range of the metrics, we applied *log normalization* [22] on Cue_e since $|prop(e)|$ can be significantly unbalanced between entity types and *min-max normalization* [23] on Cue_{er}.

3.2 $Focus_k$

Following the *KBSs focus* notion we introduced in Section 2, we model the *KBSs focus*, namely $Focus_k$, as follows:

$$Focus_k(K) = Cue_k(K) + \mu Cue_{kr}(K) = Cue_k(K)(1 + \frac{\mu}{|prop(K)|}) \ , \ \mu > 0 \quad (6)$$

where we take K as an input KBS and we take $Focus_k$ as a summation of Cue_k and Cue_{kr}. Cue_k represents the *cue validity* of the KBS. Cue_{kr} represents a normalization of Cue_k. μ represents a constraint factor for Cue_{kr}, which we assume being equal to 1, as for Cue_{er} above. $|prop(K)|$ refers to the number of the properties in K.

The notions and terminology used for entity types, i.e., the notions of Cue_e and Cue_{er}, can be straightforwardly generalized to KBSs, generating the following metrics:

$$Cue_k(K) = \sum_{i=1}^{|E_K|} Cue_e(e_i) \in [0, |prop(K)|] \quad (7)$$

The $Cue_k(K)$ is calculated as a summation of the *cue validity* of all the entity types in a given KBS, which in the function is represented by E_K. $|prop(K)|$ refers to the number of the properties in the KBS, as the maximization of Cue_k.

Following the formalization of Cue_{er} we capture the minimization level of the number of properties shared across the entity types inside the schema with the notion of Cue_{kr}, which we define as:

$$Cue_{kr}(K) = Cue_k(K) / \sum_{i=1}^{|E_K|} prop(e_i) \in [0, 1] \quad (8)$$

Cue_k and Cue_{kr} can be used then to assess the focus of a whole KBS. Notice that to normalize the metric $Focus_k$, we applied *log normalization* on Cue_k, since $|prop(K)|$ may be significantly higher in some KBSs than others KBSs and *min-max normalization* on Cue_{kr}.

4 Ranking KBSs

We started to put into use the above metrics by measuring the focus of state-of-the-art KBSs.

We collected a data set of 700 KBSs, expressed in the *Terse RDF Triple Language (Turtle)*[9] format. Most of these resources have been taken from the

[9] `https://www.w3.org/TR/turtle/`

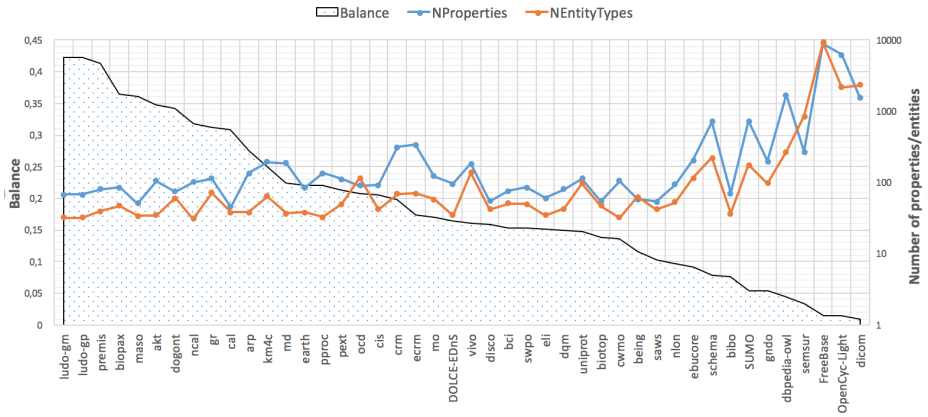

Fig. 1. *KBSs selected for the analysis*

LOV catalog[10]. The remaining ones, see for instance *freebase*[11] and *SUMO*[12] have been added to collect more data.

For the sake of the analysis, all the data sets have been flattened into a set of sets of triples (one set per entity type, or etype), where each triple encodes information about "etype-property" associations $I_K(e)$ (e.g., the triple "Person-domainOf-friend" encodes the "Person-friend" $I_K(e)$ association). Moreover, in order to generate the final output data sets we processed properties labels via NLP pipeline which performs various steps, including, for instance: *i).* split a string every time a capital letter is encountered (e.g., *birthDate* $\rightarrow$ birth and date); *ii).* lower case all characters; *iii).* filter out stop-words (e.g., *hasAuthor* $\rightarrow$ author). This allowed us to run a more accurate analysis. For instance, if "Person" and "Place" have properties like "globalLocationNumberInfo" and "LocationNumber", respectively, by processing the labels as we have done, it is possible to find some overlapping (see "location" and "number") otherwise no.

We selected a subset of the starting data set after the above processing, by discharging all the KBSs with less than 30 entity types. An overall view of the final output data set is provided by Fig.1, where, for each of the remaining 44 KBSs, the number of properties, the number of entity types, and the balance are provided. The balance returns the value of a simple distribution of the properties of a KBS across its entity types and it is calculated as:

$$Balance(K) = \frac{|prop(K_i)|}{|E_{K_i}|} * \frac{1}{|prop(e_i)|_{max}} \tag{9}$$

with $|prop(K_i)|$ being the cardinality of the set of properties of the KBS, $|E_{K_i}|$ being the cardinality of the set of entities of the KBS and $|prop(e_i)|_{max}$ being

[10] `https://lov.linkeddata.es/dataset/lov`
[11] `https://developers.google.com/freebase`
[12] `http://www.adampease.org/OP/`

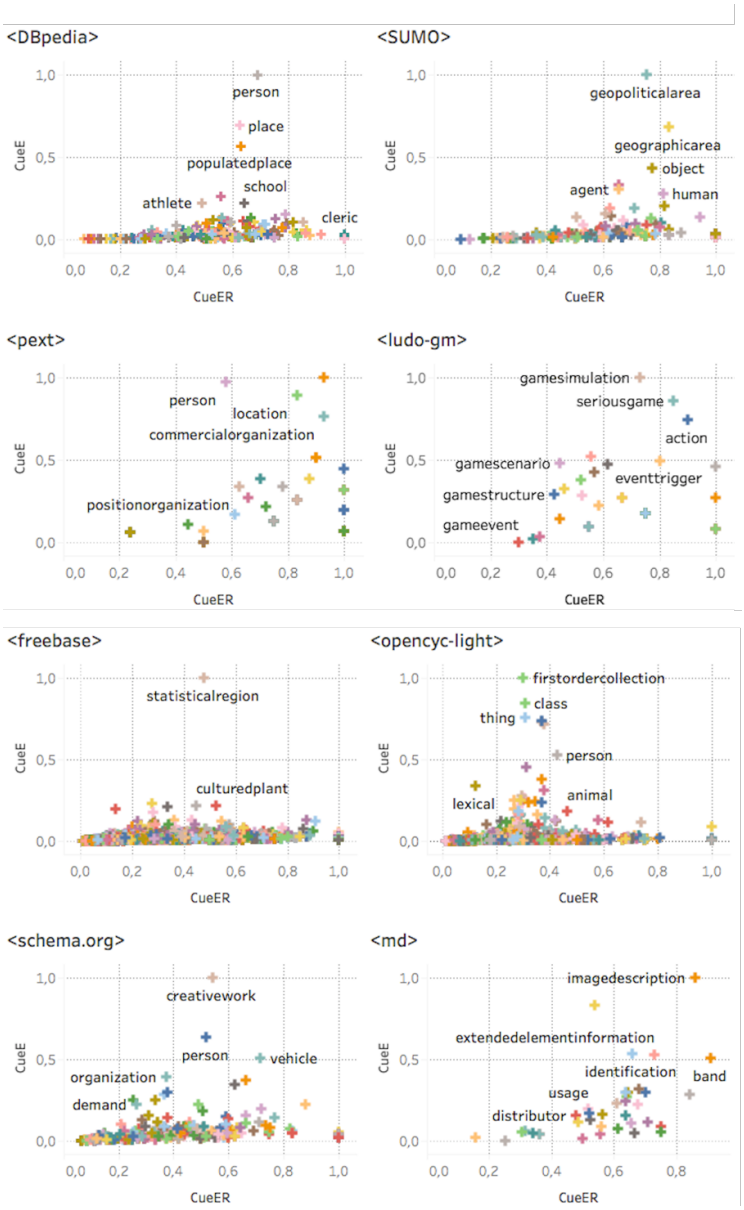

Fig. 2. *Entity types categorization relevance for eight example KBSs*

the cardinality of the set of properties associated to the entity with the major number of properties in the KBS.

By applying the cue entity metrics, i.e., $Cue_e(e)$ and $Cue_{er}(e)$ to the KBSs of the resulting list, we obtained the scores to evaluate the categorization relevance

Table 1. *KBSs ranking*

KBS	$Cue_k(K)$	$Cue_{kr}(K)$	$Focus_k(K)$
freebase:	**8981**	**0,21**	**1,15**
cal:	**46**	**0,98**	**0,92**
bibo:	**71**	**0,97**	**0,92**
opencyc-l:	**6266**	**0,26**	**0,90**
swpo:	**87**	**0,88**	**0,83**
cwmo:	**107**	**0,85**	**0,80**
eli:	**62**	**0,84**	**0,78**
ncal:	**103**	**0,80**	**0,75**
mo:	**124**	**0,79**	**0,74**
akt:	**106**	**0,79**	**0,74**

Table 2. *Entity types ranking*

KBS	**entity type**	$Cue_e(e)$	$Cue_{er}(e)$	$Focus_e(e)$
DBpedia:	*person*	**169,02**	**0,69**	**1,42**
opencyc-l:	*firstordercoll.*	**230,59**	**0,30**	**1,30**
freebase:	*statisticalreg.*	**161,53**	**0,48**	**1,17**
opencyc-l:	*class*	**194,95**	**0,31**	**1,15**
dicom:	*ieimage*	**158,90**	**0,44**	**1,13**
DBpedia:	*place*	**116,97**	**0,63**	**1,13**

Table 3. *Ranking for the entity type person from different KBSs*

KBS	**entity type**	$Cue_e(e)$	$Cue_{er}(e)$	$Focus_e(e)$
DBpedia:	*person*	**169,02**	**0,69**	**1,42**
akt:	*person*	**8,00**	**1,00**	**1,03**
opencyc-l:	*person*	**122,14**	**0,43**	**0,95**
vivo:	*person*	**10,60**	**0,88**	**0,92**
swpo:	*person*	**3,50**	**0,88**	**0,88**
cwmo:	*person*	**5,83**	**0,83**	**0,85**

of the entity types for each KBS. Let us take, for instance the values provided by KBSs in Fig.2. We randomly selected eight KBSs from the starting set and we listed them according to the number of entity types. The selected KBSs are: *Freebase*, *OpenCyc*[13], *DBpedia*, *SUMO*, *schema.org*, *md*[14], *pext*[15] and *ludo-gm*.[16]

The corresponding scattered plots provide the correlations between (a *min-max normalization* of) $Cue_e(e)$ and $Cue_{er}(e)$ for each entity type of each of the selected KBSs. The top-right entity types are the ones with the higher categorization relevance according to our metrics. For instance, in *SUMO* we have entity types like *GeopoliticalArea* and *GeographicalArea* and in *DBpedia* we have *Person* and *Place*.

By applying the $Focus_k(K)$ over the set of 44 KBSs we obtained the KBS ranking, where the top 11 KBSs are reported in Tab.1. By applying $Focus_e(e)$

[13] `https://pythonhosted.org/ordf/ordf_vocab_opencyc.html`
[14] `http://def.seegrid.csiro.au/isotc211/iso19115/2003/metadata`
[15] `http://www.ontotext.com/proton/protonext.html`
[16] `http://ns.inria.fr/ludo/v1/docs/gamemodel.html`

over the set of 44 KBSs we obtained the entity types ranking, where the top 6 entity types in terms of categorization relevance are reported in Tab.2. Finally, by selecting a given entity type, by applying $Focus_e(e)$, it is possible to find the best KBS for that entity type. Tab.3 provides an example for the entity type *Person*.

5 Validating Focus

To validate the focus metrics we use two types of assessment. In Section 5.1 we analyze the accuracy of the $Focus_e(e)$ metric in weighting the categorization relevance of entity types, namely their centrality in the maximization of information. This will be done by applying our metrics and some related SoA ranking algorithms over a set of example KBSs. Then we compare the results with a reference data set generated by 5 knowledge engineers, to which we provided a set of instructions/guidelines to rank the entity types, taking inspiration from Rosch's experiment [1]. The main goal of the assessment run in this subsection is to show how Focus reflects the judgment of engineers in measuring the relevance of a given KBS, w.r.t. a set of entity types. This also suggests a possible application of Focus in supporting search facilities in KBSs catalogs, where queries run by the users may be in the form of *"give me the most relevant KBS for the eType x and y"*.

In Section 5.2, given the lack of baseline metrics for calculating the overall score of a KBS on similar functions, and the lack of reference gold standards, we analyze the effects that the $Focus_k(K)$ of a KBS may have on the prediction performance of a relational classification task. The main goal of the assessment run in this subsection is to show how focus can support scientists in reusing KBSs in new application areas, like, for instance, *statistical relational learning* or, more precisely, in tasks like *entity type recognition*.

5.1 $Focus_e$ validation

The target here is to check how $Focus_e(e)$ allows to rank entity types in KBSs according to their categorization relevance, as described in Section 2. To assess our metric we firstly selected a subset of the KBSs discussed in the previous section, namely *akt*[17], *cwmo*[18], *ncal*, *pext*, *schema.org*, *spt*[19] and *SUMO*.

We selected these KBSs because they provide very different examples in terms of the number of properties and entity types. Moreover almost all their entity types labels are human understandable[20]. As second step we selected four SoA ranking algorithms, namely *TF-IDF* [24], *BM25* [25], *Class Match Measure (CMM)* and *Density Measure (DEM)* [26]. We used the performance of these

[17] `https://lov.linkeddata.es/dataset/lov/vocabs/akt`
[18] `https://gabriel-alex.github.io/cwmo/`
[19] `https://github.com/dbpedia/ontology-tracker/tree/master-/ontologies/spitfire-project.eu`
[20] A lot of KBSs have entity types labels codified by an ID.

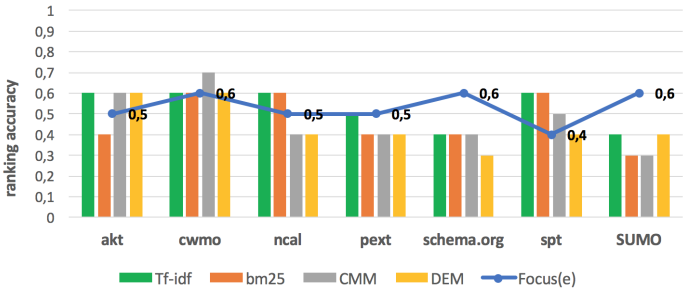

Fig. 3. $Focus_e(e)$ experiment results

rankings as a baseline, by selecting their scores for the top 10 entity types, for each of the given KBSs, and we compared them with the rankings provided by $Focus_e(e)$. The relevance of our approach was then measured in terms of accuracy (from 0 to 1) by checking how many entity types of the ranking results are in the entity types ranking lists provided by the knowledge engineers. The output of this experiment is represented by the data in Fig. 3.

As Fig.3 shows, the blue line represents the accuracy of the ranking trend provided by $Focus_e(e)$. Each bar represents the accuracy of the ranking for the corresponding selected algorithm. All the accuracy results are grouped by the reference KBS.

The first main observation is that all the reference SoA metrics show a very similar trend, with higher accuracy for *akt*, *cwmo*, *ncal* and *spt*, and lower accuracy for *schema.org* and *SUMO*. This is not the case for $Focus_e(e)$. Our metric, indeed, even if it is not the best for all the KBSs, performs best with huge and very noisy (with lower entity types Cue_{er}) KBSs, as it is the case for *schema.org* and *SUMO* (just check the visualization of *SUMO* and *schema.org* as in Figure 2 to observe the phenomenon). This, as we expected, depends on the pivotal role we gave to the minimization of the number of overlapping properties. The Cue_{er} for each entity type provides indeed essential information about the categorization relevance that, giving more importance to the number of properties of an entity type, may not be properly identified. Thus, given small and not-noisy (or "clean" in terms of number of overlapping properties) KBSs, other approaches, very focused on the number of properties of entity types pay very well (see the good performance of the *TF-IDF* algorithm). Differently, when KBSs present a huge amount of entity types, with low Cue_{er}, $Focus_e$ allows to better identify the categorization relevance.

The second main observation is that *TF-IDF* and $Focus_e(e)$ are the best metrics in terms of average performance, namely 0.52 (both *TF-IDF* and $Focus_e(e)$) mean accuracy vs. 0.47 for *bm25* and *CMM*, and 0.44 for *DEM*. This score being motivated by the fact that *TF-IDF* is almost always the best when the given KBS is small and not-noisy and $Focus_e(e)$ compensates the standard performance with small and clean KBSs, with a high performance with huge and noisy KBSs.

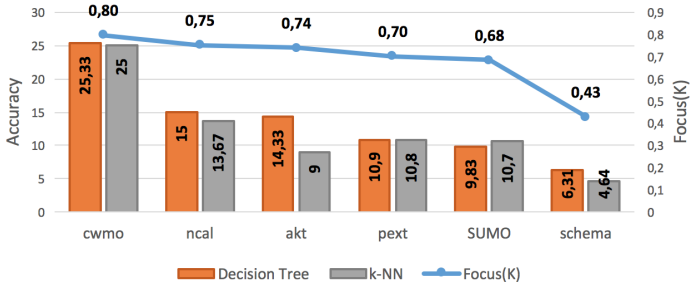

Fig. 4. $Focus_k(K)$ experiment results

5.2 $Focus_k$ validation

The target of the second task is to check whether $Focus_k(K)$ helps to predict the performance of KBSs in their ability to predict their own entity types. In this experiment, we used the same KBSs we selected in the previous experiment to address relational classification, where entity types have an associated label and the task is to predict those labels. Notice that we addressed a specific type of relational classification, namely an *entity type recognition task (ETR)*, as defined in [15]. We set-up the experiment as follows: *i).* we trained machine learning models by the FCA-format KBS as training set (In this experiment we choose *decision tree* and *k-NN* [27,28]); *ii).* we reported the relative performance of the models in terms of differences in accuracy and compared the performances with the $Focus_k(K)$ for each of the given KBSs.

As shown in Fig.4, the accuracy is reported as a proportion of correct predictions, within the range of [0%,100%]. The $Focus_k(K)$ is reported by the values of the line. The *cwmo* KBS is the one with the best scores, in terms of accuracy (for both the trained models) and $Focus_k(K)$. *schema.org* is the worst.

The main observation is that, as expected, the trend in terms of accuracy, considering both the two models, follows the $Focus_k(K)$ ranking for most of the given KBSs. However, it can be noticed that k-NN, with the *pext* KBS represents an exception, it is indeed worse than *akt* in terms of $Focus_k(K)$, but performs better with k-NN. Going deep into the analysis, this phenomenon can be explained by the relationship between the number of properties and the number of entity types, more specifically by the balance of the KBS. This value can indeed affect the performance of the model in prediction. The more the balance the more the probability of having entity types with a low focus. This effect being quite evident if we consider two KBSs with extremely similar $Focus_k(K)$, but disparate balance. This experiment, while showing how $Focus_k(K)$ can be a concrete explanation of the categorization relevance of a KBS, suggests the possibility of a practical application of $Focus_k(K)$ to measure the potential performance of a KBS or a set of KBSs in a relational classification task. The results may be used, e.g., to fine-tune KBSs in an open-world data integration scenario.

6 Related work

Our work shares with the research on *ontology* and *knowledge graph (KG) schema* (functional) evaluation [7, 9, 29] the goal of facilitating the reuse of these knowledge structures. This work has been extensive and has exploited a huge amount of methods and techniques including, e.g. *DWRank* [30] and the *NCBO* [31] (the former being a high precision recommender for biomedical ontology, the latter being a "learning to rank approach" based on search queries).

Our proposal differs from this related work in two major respects. The first is that we ground our approach and the notion of focus on the notion of categorization purpose from cognitive psychology. The theoretical underpinning of our formalization of the metrics and the experimental setup is then inspired by the analysis of human behavior in categorization, and in particular by the seminal work by E. Rosch. Our goal is not to redefine terminology already in use in the related work, but rather to propose a both theoretically and practically useful formalization of the central activity of categorization, which can be considered as the baseline of each knowledge engineering task. The second difference, which is actually a consequence of the first, is that, while most of the functional evaluation approaches are related to the intended use of a given KBS, and consider functional dimensions, like task and domain, which are very context-dependent, this is not the case with our approach. The notion of focus we adapted, indeed, aims to model a privileged level of categorization, independently from the tasks and the domain of application of the data structure. This in turn allows us to devise a somewhat opposite approach. In fact, the domain of a KBS can be then identified through the focus scores. For instance, the fact that a KBS has a high focus for entity types like *CreativeWork* or *Product*, will help the user to understand what is the real potential of that KBS for a given domain of application.

As a final remark, it is important to observe how the notion of cue validity has been widely studied in the context of feature engineering. Together with other similar measures as "category utility" or "mutual information" and, it has been used to measure the informativeness of a category [32]. Our approach differs from the related work in the application of Rosch's notion at the KBS level, rather than on data. Moreover, the introduction of the "overall" *Focus* metrics to rank categorization relevance is a novel contribution.

7 Conclusion

In this paper, we have proposed a formal method to evaluate KBSs according to their focus, namely, what cognitive psychologists call categorization purpose. This in turn has allowed us to describe how this evaluation plays an important role in supporting an accurate level of KBSs understanding and reuse.

In this regard, as preliminary validation of the proposed metrics we are showing: *a).* how focus KBSs ranking reflects the ranking of the KBSs provided by a group of knowledge engineers, following the guidelines inspired by a well-known

experiment in cognitive psychology; *b).* how focus can help scientists in selecting better KBSs to train a classifier and address an *Entity Type Recognition (ETR)* task.

The future work will concentrate on an extension of the proposed metrics, possibly by considering the hierarchical structure of KBSs, an extension of the experimental set-up, and an implementation of the metrics for supporting the search engine of a large number of existing high-quality KBSs.

Acknowledgement

The research conducted by Mattia Fumagalli is supported by the *"NEXON - Foundations of Next-Generation Ontology-Driven Conceptual Modeling"* project, funded by the *Free University of Bozen-Bolzano*. The research conducted by Fausto Giunchiglia and Daqian Shi has received funding from the *"DELPhi - DiscovEring Life Patterns"*, funded by the MIUR Progetti di Ricerca di Rilevante Interesse Nazionale (PRIN) 2017 – DD n. 1062.

References

1. E. Rosch. Principles of categorization. *Concepts: core readings*, 189, 1999.
2. Stevan Harnad. To cognize is to categorize: Cognition is categorization. In *Handbook of categorization in cognitive science*, pages 21–54. Elsevier, 2017.
3. Ruth Garrett Millikan. *On clear and confused ideas: An essay about substance concepts*. Cambridge University Press, 2000.
4. Fausto Giunchiglia and Mattia Fumagalli. Concepts as (recognition) abilities. In *FOIS*, pages 153–166, 2016.
5. Liu Qiao, Li Yang, Duan Hong, Liu Yao, and Qin Zhiguang. Knowledge graph construction techniques. *Journal of computer research and development*, 53(3):582–600, 2016.
6. Nicola Guarino, Daniel Oberle, and Steffen Staab. What is an ontology? In *Handbook on ontologies*, pages 1–17. Springer, 2009.
7. Heiko Paulheim. Knowledge graph refinement: A survey of approaches and evaluation methods. *Semantic web*, 8(3):489–508, 2017.
8. Nigel Shadbolt, Paul R Smart, JR Wilson, and S Sharples. Knowledge elicitation. *Evaluation of human work*, pages 163–200, 2015.
9. Melinda McDaniel and Veda C Storey. Evaluating domain ontologies: Clarification, classification, and challenges. *ACM Computing Surveys (CSUR)*, 52(4):1–44, 2019.
10. Auriol Degbelo. A snapshot of ontology evaluation criteria and strategies. In *Proceedings of the 13th International Conference on Semantic Systems*, pages 1–8, 2017.
11. Maximilian Nickel, Kevin Murphy, Volker Tresp, and Evgeniy Gabrilovich. A review of relational machine learning for knowledge graphs. *Proceedings of the IEEE*, 104(1):11–33, 2015.
12. Mattia Fumagalli, Gábor Bella, Samuele Conti, and Fausto Giunchiglia. Ontology-driven cross-domain transfer learning. In *Formal Ontology in Information Systems*, pages 249–263. IOS Press, 2020.

13. Ruth Garrett Millikan. *Beyond concepts: Unicepts, language, and natural information*. Oxford University Press, 2017.
14. Birger Hjørland and Frank Sejer Christensen. Work tasks and socio-cognitive relevance: A specific example. *Journal of the American Society for Information Science and Technology*, 53(11):960–965, 2002.
15. Fausto Giunchiglia and Mattia Fumagalli. Entity type recognition–dealing with the diversity of knowledge. In *Proceedings of the International Conf. on Principles of Knowledge Representation and Reasoning*, volume 17, pages 414–423, 2020.
16. Mattia Fumagalli, Gábor Bella, and Fausto Giunchiglia. Towards understanding classification and identification. In *Pacific Rim International Conference on Artificial Intelligence*, pages 71–84. Springer, 2019.
17. Eleanor Rosch and Carolyn B Mervis. Family resemblances: Studies in the internal structure of categories. *Cognitive psychology*, 7(4):573–605, 1975.
18. Fausto Giunchiglia and Mattia Fumagalli. On knowledge diversity. *Proceedings of the 2019 Joint Ontology Workshops, WOMoCoE*, 2019.
19. B. Ganter and R. Wille. *Formal concept analysis: mathematical foundations*. Springer, 2012.
20. Palash Goyal and Emilio Ferrara. Graph embedding techniques, applications, and performance: A survey. *Knowledge-Based Systems*, 151:78–94, 2018.
21. Fausto Giunchiglia and Mattia Fumagalli. On knowledge diversity. In *JOWO*, 2019.
22. E Bornemann, JH Doveton, et al. Log normalization by trend surface analysis. *The log analyst*, 22(04), 1981.
23. Y Kumar Jain and Santosh Kumar Bhandare. Min max normalization based data perturbation method for privacy protection. *International Journal of Computer & Communication Technology*, 2(8):45–50, 2011.
24. Gerard Salton and Christopher Buckley. Term-weighting approaches in automatic text retrieval. *Information processing & management*, 24(5):513–523, 1988.
25. Stephen E Robertson, Steve Walker, Susan Jones, Micheline M Hancock-Beaulieu, Mike Gatford, et al. Okapi at trec-3. *Nist Special Publication Sp*, 109:109, 1995.
26. Harith Alani and Christopher Brewster. Metrics for ranking ontologies. 2006.
27. Bogumił Kamiński, Michał Jakubczyk, and Przemysław Szufel. A framework for sensitivity analysis of decision trees. *Central European journal of operations research*, 26(1):135–159, 2018.
28. Belur V Dasarathy. Nearest neighbor (nn) norms: Nn pattern classification techniques. *IEEE Computer Society Tutorial*, 1991.
29. Aldo Gangemi, Carola Catenacci, Massimiliano Ciaramita, and Jos Lehmann. A theoretical framework for ontology evaluation and validation. In *SWAP*, volume 166, page 16. Citeseer, 2005.
30. Anila Sahar Butt, Armin Haller, and Lexing Xie. Dwrank: Learning concept ranking for ontology search. *Semantic Web*, 7(4):447–461, 2016.
31. Marcos Martínez-Romero, Clement Jonquet, Martin J O'connor, John Graybeal, Alejandro Pazos, and Mark A Musen. Ncbo ontology recommender 2.0: an enhanced approach for biomedical ontology recommendation. *Journal of biomedical semantics*, 8(1):21, 2017.
32. Hanchuan Peng, Fuhui Long, and Chris Ding. Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (8):1226–1238, 2005.