Semantic Search via Entity-Types: The SEMANNOREX Framework

Amit Kumar
Université de Caen Normandie
Caen, France
amit.kumar@unicaen.fr

Govind
Université de Caen Normandie
Caen, France
govind@unicaen.fr

Marc Spaniol
Université de Caen Normandie
Caen, France
marc.spaniol@unicaen.fr

ABSTRACT

Capturing and exploiting a content's semantic is a key success factor for Web search. To this end, it is crucial to - ideally automatically - extract the core semantics of the data being processed and link this information with some formal representation, such as an ontology. By intertwining both, search becomes semantic by simultaneously allowing end-users a structured access to the data via the underlying ontology. Connecting both, we introduce the SEMANNOREX framework in order to provide semantically enriched access to a news corpus from Websites and Wikinews.

KEYWORDS

Entity-level Analytics, Semantic Search via Entity-types

ACM Reference Format:
Amit Kumar, Govind, and Marc Spaniol. 2021. Semantic Search via Entity-Types: The SEMANNOREX Framework. In *Companion Proceedings of the Web Conference 2021 (WWW '21 Companion), April 19–23, 2021, Ljubljana, Slovenia.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3442442.3458607

1 INTRODUCTION

Collaborative tagging has been widely established as a method of content annotation and retrieval since the beginning of the Web 2.0 era [13]. Applications range from tagging of books[1], via annotations of songs[2], up to editorial contents provided in commercial platforms[3]. To this end, tagging requires qualified human annotators producing a "bag of tags" content annotation. The result is a flat model that isn't capable of exploiting the inherent semantic dependencies associated with each tag, e.g., the similarity between an ATHLETE and a PLAYER. However, the proliferation of linked open data (LOD) and knowledge bases (KBs) such as DBpedia [1] or YAGO [20], allows making those dependencies expressible and measurable. In order to overcome the shortcoming of relying onto high-quality manual annotations within a "bag of tags" representation, we present the SEMANNOREX (SEMantic ANNOtation, Retrieval and EXploration) framework for semantic search via entity-types.

[1]https://blog.librarything.com/main/category/tags/
[2]http://www.deezer-blog.com/tags-in-search/
[3]https://www.bbc.co.uk/blogs/aboutthebbc/tags

ACM ISBN 978-1-4503-8313-4/21/04.
https://doi.org/10.1145/3442442.3458607

2 CONCEPTUAL APPROACH

2.1 Document Collection

The conceptual approach of SEMANNOREX is shown in Figure 1. It builds upon more than 400 types structured by the 5 top-level types from the YAGO ontology [20]. In our demo, we utilize an English corpus of Web news contents and Wikinews[4] (cf. ① in Fig. 1).

2.2 Semantic Annotation

The semantic annotation is obtained from the named entities present in the document. These named entities in the Web contents can be identified by employing a named entity disambiguation tool [9, 16, 21]. For SEMANNOREX, we employ AIDA-light [17] for disambiguation of Web news contents as well as mapping linked Wikipedia pages onto the canonicalized YAGO [8, 20] entity for Wikinews data (cf. ② in Fig. 1). Since KBs capture plenitude of information about named entities via the transitive closure (e.g. in YAGO 42 types for *Emmanuel Macron* or 14 for the *European Banking Authority (EBA)*), we focus on the most "representative" type(s) by employing the PURE framework [12] (cf. ③ in Fig. 1).

2.3 Semantic Retrieval & Exploration

For retrieval we allow three different methods (cf. ④ in Fig. 1). We define q as the user query types and d the types of an annotated document, where q_{τ_i} and d_{τ_j} stands for the types present in the query and the document, respectively.

$$q = \{q_{\tau_1}, q_{\tau_2} \ldots q_{\tau_i}\} \text{ and } d = \{d_{\tau_1}, d_{\tau_2} \ldots d_{\tau_j}\}$$

Here, a non-zero value indicates the presence of the type. The computation is then based on the vectors for the query $\Pi(q)$ and the document $\Pi(d)$.

Cosine Similarity
The document vector entries are assigned as the number of times a type appears in the same document. The computation of cosine similarity (cf. [14]) is defined as:

$$cos(\Pi(d), \Pi(q)) = {(\Pi(d) \cdot \Pi(q))}/{(\|\Pi(d)\| \, \|\Pi(q)\|)}$$

Semantic Pathlength
In order to incorporate the structure of underlying ontology, we also utilize the Pathlength [10, 19] as measure of semantic similarity defined as follows:

$$sempath(q, d) = avg_{1 \le m \le i} \left(\max_{1 \le n \le j} \left(\frac{1}{1 + pathlength(q_{\tau_m}, d_{\tau_n})} \right) \right)$$

Semantic Content Similarity (*SCS*) of KB Types
In *SCS* we adopt the Resnik approach [18] of assessing type similarity within our ontology. To this end, we treeify the directed acyclic

[4]https://spaniol.users.greyc.fr/research/SEMANNOREX/SEMANNOREX.zip

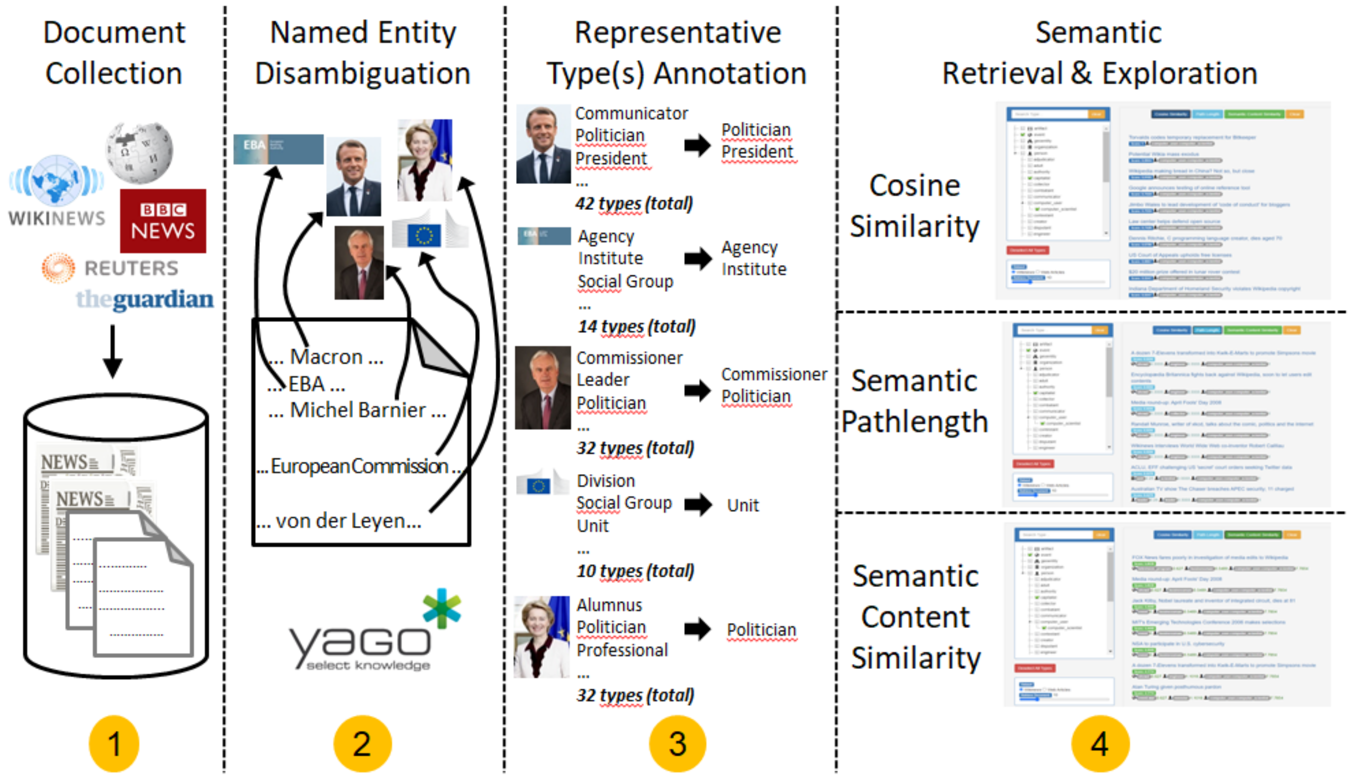

Figure 1: Conceptual SEMANNOREX Pipeline

graph (DAG) of the YAGO ontology by (recursively) duplicating child nodes having multiple parent nodes in each parent's (sub-) branch (cf. Figure 2). As a consequence of treeification, content types annotations are classified in the (sub-)branch associated with the parent node of the "predominant" top-level type. This means, the "duplicated type" will be linked only to that parent node, which belongs to the top-level type where the majority of the remaining types of this content belong to. In case, where the majority voting leads to a draw, the content will be typed to each of these duplicated types. The pseudo code of the ontology treeification process is presented in Algorithm 1.

Let $\hat{\tau}_i$ be the set of all the successor types of τ_i and itself. Then, we compute for each type τ_i its probability, defined as:

$$P(\tau_i) = \frac{\sum_{\tau \in \hat{\tau}_i} count(\tau)}{N}$$

Here, N is the frequency of total types and $count(\tau)$ is frequency of type τ. Let $LCA(\tau_x, \tau_y)$ be the lowest common ancestor of types τ_x and τ_y, then SCS is:

$$SCS(\tau_x, \tau_y) = -logP(LCA(\tau_x, \tau_y))$$

$$SCS(q, d) = avg\left(\max_{1 \leq n \leq j} SCS(q_{\tau_m}, d_{\tau_n})\right), 1 \leq m \leq i$$

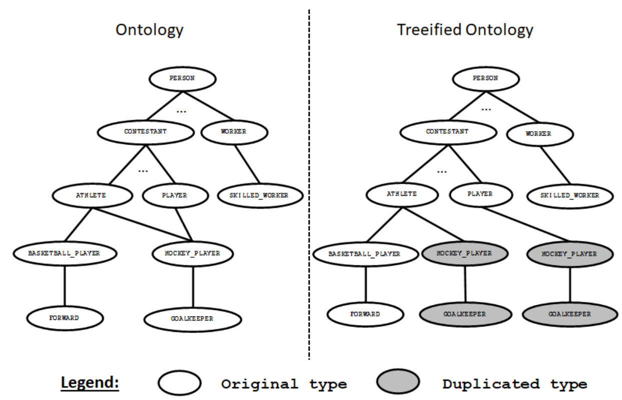

Figure 2: Ontology Treeification

3 SEMANNOREX DEMONSTRATION

The SEMANNOREX demo showcases semantic search via entity-types based on **Cosine Similarity**, **Semantic Pathlength** as well as **Semantic Content Similarity** on a corpus of Web news and Wikinews articles. Figure 3 depicts the different retrieval strategies,

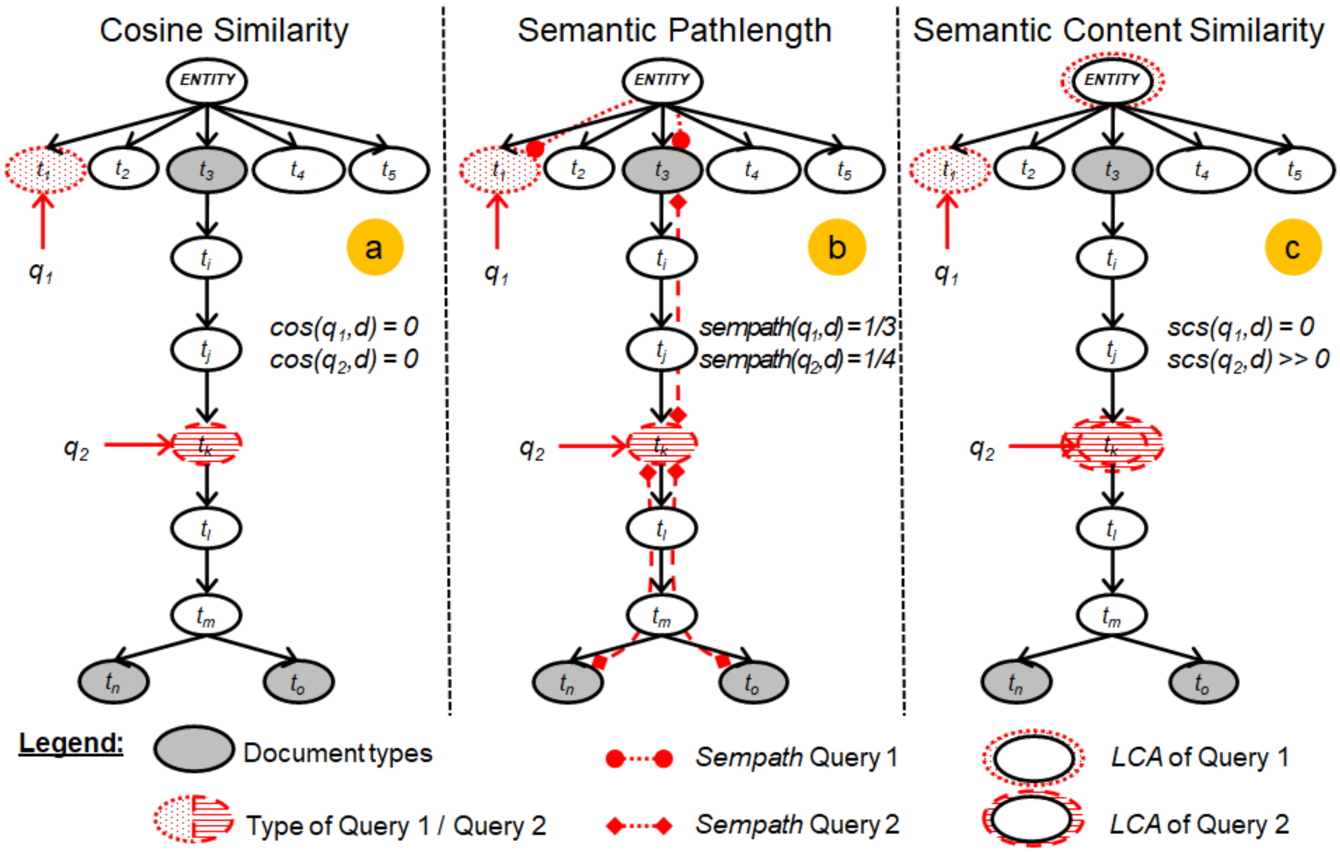

Figure 3: Comparison of the different Retrieval Methods

Algorithm 1 Ontology Treeification

Input: Original Ontology ($\mathcal{T} = t_1, t_2, \ldots, t_I$);
PARENTS(t) returns parents of node t;
len(*PARENTS*(t)) returns number of parents of node t;
CHILDREN(t) returns all the children of node t;
CHILDADD(t, p) sets node p as one of the children of node t;
REMOVE(t) deletes the subtree rooted at node t

Output: Treeified Ontology

```
for t_i ∈ 𝒯 do
    if len(PARENTS(t_i)) > 1 then
        for p ∈ PARENTS(t_i) do
            t_i_new ← p + "." + t_i
            CHILDADD(p, t_i_new)
            for child ∈ CHILDREN(t_i) do
                t_i_new_child ← t_i_new + "." + child
                CHILDADD(t_i_new, t_i_new_child)
        REMOVE(t_i)
return 𝒯
```

which will be presented subsequently. A demonstration video and a live demonstrator can be found at the SEMANNOREX Website[5].

[5]https://spaniol.users.greyc.fr/research/SEMANNOREX/

Cosine Similarity (Cosine)
Cosine Similarity serves as a "baseline" retrieval method. The user might experience a somewhat "extreme" system behavior whether the selected type is present in the document, or not. This is due to the fact, that type vectors of documents tend to be sparse and semantic dependencies such as parent-child or sibling relations can not be exploited for retrieval. As a result, both sample queries in Fig. 3 ⓐ do not return the document labeled by the grey types.

Semantic Pathlength (SemPath)
Semantic Pathlength aims at overcoming the above mentioned shortcomings, through capturing parent-child or sibling relations by considering the distance between the selected type(s) and its (their) best possible match(es) in the document(s). However, the main drawback now is that types in the upper part of the ontology by definition are relatively "close" to the remaining types. Thus, example query q_1 scores higher than q_2 in Fig. 3 ⓑ, although all document types are in same branch of query q_2 while q_1 belongs to a different top-level type.

Semantic Content Similarity (SCS) of KB Types
Finally, Semantic Content Similarity (SCS) allows to exploit the semantics inherent in parent-child or sibling relations as well as putting "emphasis" on more specific types. To this end, the impact

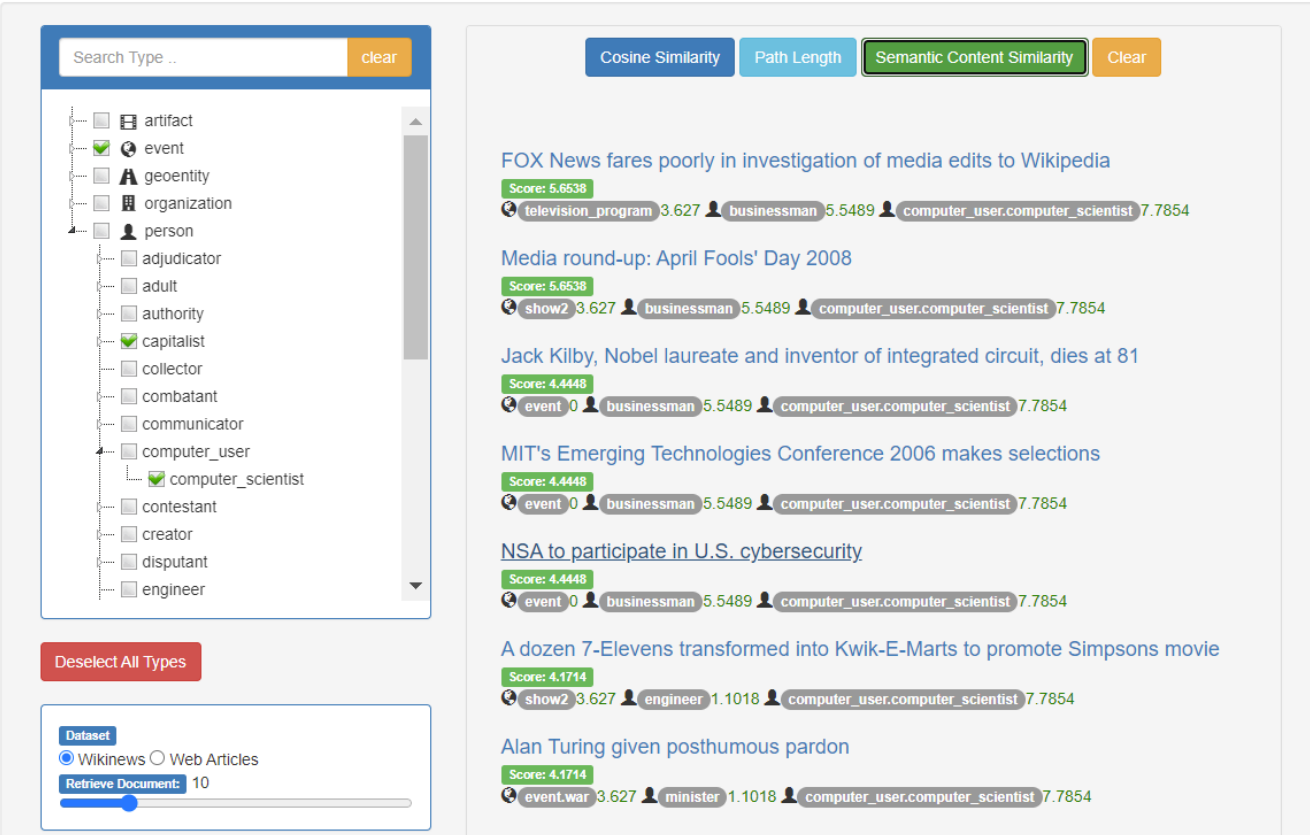

Figure 4: SEMANNOREX Search Interface displaying Results based on the Semantic Content Similarity (SCS) Method

of an *LCA* type at the lower part of the ontology will be higher compared with an *LCA* type at a higher part of the ontology and, thus, leading to more concise search results. In the example of Fig. 3 ©, now, query q_1 does not return the document containing the grey types, because the *LCA* is the *root* node. In contrast, for q_2 the document will achieve a comparatively high score as the *LCA* of query and document is type t_k.

SEMANNOREX Search Interface

Figure 4 depicts the user interface of SEMANNOREX showing an example query for the types `computer_scientist`, `capitalist` and `event`. On the left hand side, the corpus (Wikinews or Web news) can be selected [at the bottom] and the treeified ontology can be explored [on top]. From the ontology representation one or more types of interest can be selected. The search results are retrieved and ranked accordingly in the main panel of the interface. In this example, the results are shown for the Semantic Content Similarity (SCS) method. In order to allow the user an intuition about the linked content, its title and the scores per selected type are provided. Further, the buttons on top allow the user to alter the utilized scoring method. Thus, the user is able to assess and compare the relevance of the documents listed with respect to the individual types as well as based on the underlying scoring method.

Evaluation

The demo corpus exists of more than 22,000 Web news and Wikinews articles. Table 1 summarizes the findings mentioned above conducted on 50 manually assessed queries each on Web news as well as on Wikinews articles. These queries range from 1 to 5 randomly chosen entity type(s), thus, emulating search behavior of various complexity. In order to ensure comparability, 10 queries have been constructed for each "level" (i.e. 10 queries with one type, two types, etc.). It can be observed from Table 1 that SCS ensures a balance between scarcity and information overload by simultaneously achieving the highest quality in terms of Prec@5 and MRR.

Method	Quantitative				Qualitative	
	Min	Max	Avg.	Median	Prec@5	MRR
`Cosine`	0	6,629	511.71	118.5	0.499	0.558
`SemPath`	3,662	18,929	11,295.5	11,295.5	0.590	0.711
`SCS`	1,417	18,903	8,653.47	5,281	0.641	0.771

Table 1: Quantitative and Qualitative Evaluation

In addition, we present the analysis of a sensitivity study in Table 2. It can be observed that the results for Cosine are somewhat extreme: queries with few entity types (one or two) lead to highly concise results (in case they exist), while a decay in quality can be observed for queries with more entity types. This observed decay can be dampened by the two other methods incorporating the underlying ontology structure (SemPath and SCS). Here, SCS is overall performing better. This is primarily caused by the fact that SemPath does establish links to all documents in the corpus (cf. quantitative analysis of Table 2) and, thus, also retrieves documents that are conceptually quite dissimilar. In contrast, SCS is more focused and retrieves only those documents that belong to the same top-level type. As a result, the number of documents retrieved is less, but they are overall more relevant.

Method	# of Types	Quantitative				Qualitative	
		Min	Max	Avg.	Median	Prec@5	MRR
Cosine	1	0	2,565	463.6	25	0.707	0.695
	2	0	386	84.9	43	0.75	0.589
	3	3	995	210.6	98	0.554	0.675
	4	17	6,629	1,197.75	538.5	0.437	0.618
	5	51	3,542	601.7	402	0.165	0.214
SemPath	1	3,662	18,929	11,295.5	11,295.5	0.73	0.842
	2	3,662	18,929	11,295.5	11,295.5	0.642	0.77
	3	3,662	18,929	11,295.5	11,295.5	0.482	0.607
	4	3,662	18,929	11,295.5	11,295.5	0.632	0.721
	5	3,662	18,929	11,295.5	11,295.5	0.462	0.617
SCS	1	1,417	16,644	7,256.95	5,161	0.72	0.87
	2	1,656	18,025	8,868.8	5,897.5	0.682	0.837
	3	2,959	18,747	8,592.7	7,123.5	0.627	0.731
	4	2,959	18,478	9,115.15	7,112.5	0.686	0.854
	5	2,959	18,903	9,433.75	7,124,5	0.49	0.568

Table 2: Sensitivity Study

4 RELATED WORK

Work on automatic classification of documents with predefined types/categories has been studied in [11]. GoNTogle [2, 3] supports semantic and keyword-based search over documents. However, none of the systems is solely built upon entity related information. STICS [7] aims at semantic annotation and retrieval via named entities, but does not exploit conceptual or structural similarity. CALVADOS [6] enables content summarization on semantic level via semantic fingerprinting [4, 5], but does not support content retrieval or exploration. TagTheWeb [15] tags documents based on taxonomic relations in Wikipedia, but does neither provide a proper search interface nor exploit semantic similarity of concepts/tags.

5 CONCLUSIONS & OUTLOOK

This demo presented SEMANNOREX, a novel tool for the semantic annotation, retrieval and exploration of (textual) documents. The novelty arises from exploiting concise entity-level annotations for semantic retrieval. As a proof-of-concept implementation, we applied SEMANNOREX onto a news corpus collected from Websites and Wikinews. In future, we intend to apply SEMANNOREX onto additional datasets.

In particular, we aim at applying SEMANNOREX in the context of the ASTURIAS (Analyse STructURelle et Indexation sémantique d'ArticleS de presse) project onto a large, digitized corpus of French news articles. By doing so, we intend to provide the end-user with an innovative semantically-driven access paradigm in order to explore (textual) document archives.

ACKNOWLEDGEMENTS

This work was supported by the RIN RECHERCHE Normandie Digitale research project ASTURIAS contract no. 18E01661. We thank our colleagues for the inspiring discussions.

REFERENCES

[1] Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. 2007. DBpedia: A Nucleus for a Web of Open Data. In *ISWC/ASWC*. 722–735.

[2] Nikos Bikakis, Giorgos Giannopoulos, Theodore Dalamagas, and Timos Sellis. 2010. Integrating Keywords and Semantics on Document Annotation and Search. In *On the Move to Meaningful Internet Systems, OTM 2010*. Springer Berlin Heidelberg, Berlin, Heidelberg, 921–938.

[3] Giorgos Giannopoulos, Nikos Bikakis, Theodore Dalamagas, and Timos Sellis. 2010. GoNTogle: A Tool for Semantic Annotation and Search. In *The Semantic Web: Research and Applications*. Springer, Berlin, Heidelberg, 376–380.

[4] Govind, Céline Alec, and Marc Spaniol. 2018. Semantic Fingerprinting: A Novel Method for Entity-Level Content Classification. In *Web Engineering - 18th International Conference, ICWE 2018, Cáceres, Spain, June 5-8, 2018, Proceedings*. 279–287.

[5] Govind, Céline Alec, and Marc Spaniol. 2019. Fine-grained Web Content Classification via Entity-level Analytics: The Case of Semantic Fingerprinting. *Journal of Web Engineering (JWE)* 17, 6&7 (2019), 449–482.

[6] Govind, Amit Kumar, Céline Alec, and Marc Spaniol. 2019. CALVADOS: A Tool for the Semantic Analysis and Digestion of Web Contents. In *Proc. of the* 16^{th} *Extended Semantic Web Conference (ESWC 2019), Portorož, Slovenia, June 2-6*. 84–89.

[7] Johannes Hoffart, Dragan Milchevski, and Gerhard Weikum. 2014. STICS: Searching with strings, things, and cats. https://doi.org/10.1145/2600428.2611177

[8] Johannes Hoffart, Fabian M. Suchanek, Klaus Berberich, and Gerhard Weikum. 2013. YAGO2: A spatially and temporally enhanced knowledge base from Wikipedia. *Artificial Intelligence* 194 (2013), 28–61.

[9] Johannes Hoffart, Mohamed Amir Yosef, Ilaria Bordino, Hagen Fürstenau, Manfred Pinkal, Marc Spaniol, Bilyana Taneva, Stefan Thater, and Gerhard Weikum. 2011. Robust Disambiguation of Named Entities in Text. In *Conference on Empirical Methods in Natural Language Processing, Edinburgh, Scotland, UK*. 782–792.

[10] Baoxian Jia, Xin Huang, and Shuang Jiao. 2018. Application of Semantic Similarity Calculation Based on Knowledge Graph for Personalized Study Recommendation Service. *Educational Sciences: Theory & Practice*, 18(6) (2018), 2958–2966.

[11] Rajni Jindal, Ruchika Malhotra, and Abha Jain. 2015. Techniques for text classification: Literature review and current trends. *Webology* 12 (2015).

[12] Amit Kumar, Govind, Céline Alec, and Marc Spaniol. 2020. Blogger or President? Exploitation of Patterns in Entity Type Graphs for Representative Entity Type Classification. In *Proc. of the 12th Intl. ACM Web Science Conference (WebSci '20)*. 59–68.

[13] George Macgregor and Emma Mcculloch. 2006. Collaborative tagging as a knowledge organisation and resource discovery tool. *Library Review* 55, 5 (2006), 291–300.

[14] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval*. Cambridge University Press, USA.

[15] Jerry Fernandes Medeiros, Bernardo Pereira Nunes, Sean Wolfgand Matsui Siqueira, and Luiz André Portes Paes Leme. 2018. TagTheWeb: Using Wikipedia Categories to Automatically Categorize Resources on the Web. In *ESWC (Satellite Events) (Lecture Notes in Computer Science, Vol. 11155)*. Springer, 153–157.

[16] Pablo N. Mendes, Max Jakob, Andrés García-Silva, and Christian Bizer. 2011. DBpedia Spotlight: Shedding Light on the Web of Documents. In *Proceedings of the 7th International Conference on Semantic Systems* (Graz, Austria) *(I-Semantics '11)*. ACM, New York, NY, USA, 1–8.

[17] Dat Ba Nguyen, Johannes Hoffart, Martin Theobald, and Gerhard Weikum. 2014. AIDA-light: High-throughput named-entity disambiguation. *CEUR Workshop Proceedings* 1184.

[18] Philip Resnik. 1995. Using Information Content to Evaluate Semantic Similarity in a Taxonomy. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 1* (Montreal, Quebec, Canada) *(IJCAI '95)*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 448–453.

[19] Thabet Slimani. 2013. Description and evaluation of semantic similarity measures approaches. *arXiv preprint arXiv:1310.8059* (2013).

[20] Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2007. YAGO: A Core of Semantic Knowledge - Unifying WordNet and Wikipedia. In *16th International World Wide Web Conference (WWW 2007)*. ACM, 697–706.

[21] Mohamed Amir Yosef, Johannes Hoffart, Ilaria Bordino, Marc Spaniol, and Gerhard Weikum. 2011. AIDA: An Online Tool for Accurate Disambiguation of Named Entities in Text and Tables. *Proc. VLDB Endow.* 4 (2011), 1450–1453.