applied sciences

MDPI

Article

Wikidata Support in the Creation of Rich Semantic Metadata for Historical Archives

Davide Colla, Annamaria Goy *, Marco Leontino and Diego Magro

Dipartimento di Informatica, Università di Torino, 10149 Torino, Italy; davide.colla@unito.it (D.C.); marco.leontino@unito.it (M.L.); diego.magro@unito.it (D.M.)
* Correspondence: annamaria.goy@unito.it

Abstract: The research question this paper aims at answering is the following: In an ontology-driven annotation system, can the information extracted from external resources (namely, Wikidata) provide users with useful suggestions in the characterization of entities used for the annotation of documents from historical archives? The context of the research is the PRiSMHA project, in which the main goal is the development of a proof-of-concept prototype ontology-driven system for semantic metadata generation. The assumption behind this effort is that an effective access to historical archives needs a rich semantic knowledge, relying on a domain ontology, that describes the content of archival resources. In the paper, we present a new feature of the annotation system: when characterizing a new entity (e.g., a person), some properties describing it are automatically pre-filled in, and more complex semantic representations (e.g., events the entity is involved in) are suggested; both kinds of suggestions are based on information retrieved from Wikidata. In the paper, we describe the automatic algorithm devised to support the definition of the mappings between the Wikidata semantic model and the PRiSMHA ontology, as well as the process used to extract information from Wikidata and to generate suggestions based on the defined mappings. Finally, we discuss the results of a qualitative evaluation of the suggestions, which provides a positive answer to the initial research question and indicates possible improvements.

Keywords: semantic metadata generation; linked data; ontologies; digital curation; historical archives; digital cultural heritage; information exploration

check for updates

Citation: Colla, D.; Goy, A.; Leontino, M.; Magro, D. Wikidata Support in the Creation of Rich Semantic Metadata for Historical Archives. *Appl. Sci.* **2021**, *11*, 4378. https://doi.org/10.3390/app11104378

Academic Editor: Liliana Ardissono

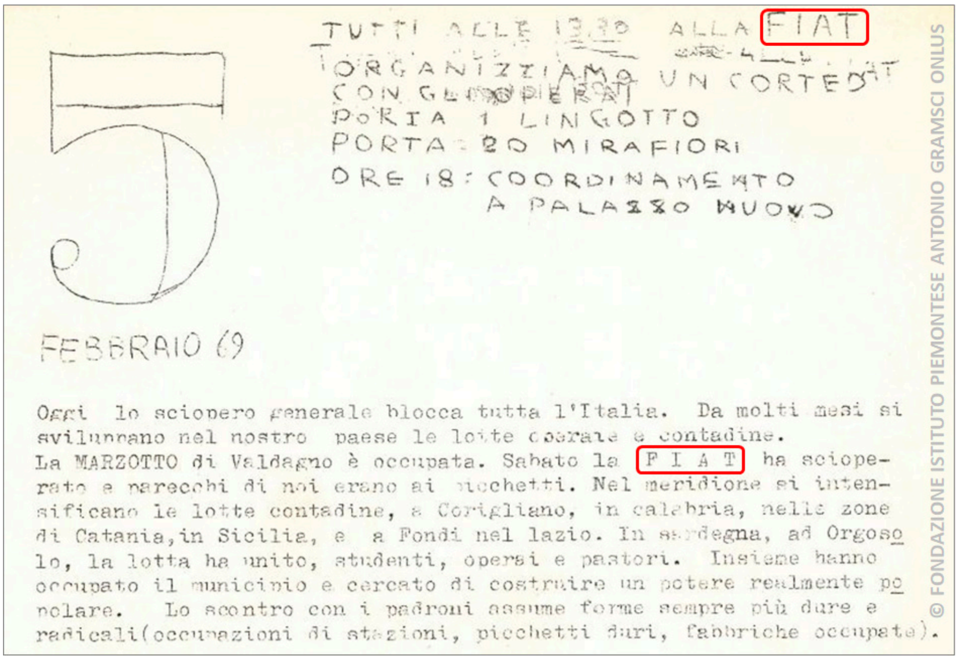
Received: 9 April 2021
Accepted: 6 May 2021
Published: 12 May 2021

1. Introduction

Digital Cultural Heritage has become a key concept for all kinds of cultural institutions, whether they are museums, libraries, archives, or cultural centers. It is a pillar of EU research programs (in both Horizon2020 and Horizon Europe frameworks: ec.europa.eu/info/research-and-innovation/research-area/social-sciences-and-humanities/europes-cultural-heritage-and-creativity_en); it represents the core of many European initiatives (pro.europeana.eu/post/understanding-digital-transformation-across-the-cultural-heritage-sector); it attracted the specific attention of UNESCO, who claims: "Digital heritage is likely to become more important and more widespread over time. Increasingly, individuals, organizations and communities are using digital technologies to document and express what they value and what they want to pass on to future generations". (en.unesco.org/themes/information-preservation/digital-heritage/concept-digital-heritage) Obviously, Cultural Heritage is a very inclusive concept, and different types of cultural institutions have heterogeneous needs with respect to the management of their Cultural Heritage in a digital framework. Moreover, the term "digital" itself has a large and sometimes fuzzy meaning, ranging from the digitization of paper documents to virtual reality applications, to online communication and marketing activities.

In this paper, we will focus on cultural institutions that host *historical archives* including documents such as newspaper articles, pictures, typewritten leaflets, manuscripts, and posters.

The digital transformation for historical archives is particularly challenging: The simple digitization of documents is not enough to offer an effective and flexible access, since the actual content of documents must be grasped. However, a completely automatic processing aimed at extracting document content can be very difficult, sometimes already at the OCR level (e.g., in case of very blurred old text with handwritten annotations—see Figure 1), sometimes when trying to apply complex NLP approaches (e.g., event-extraction [1]) to OCR-ized texts.



TUTTI ALLE 13.30 ALLA FIAT
ORGANIZZIAMO UN CORTEO
CON GLI OPERAI
PORTA 1 LINGOTTO
PORTA 20 MIRAFIORI
ORE 18: COORDINAMENTO
A PALAZZO NUOVO

5
FEBBRAIO 69

Oggi lo sciopero generale blocca tutta l'Italia. Da molti mesi si sviluppano nel nostro paese le lotte operaie e contadine. La MARZOTTO di Valdagno è occupata. Sabato la FIAT ha scioperato e parecchi di noi erano ai picchetti. Nel meridione si intensificano le lotte contadine, a Corigliano, in calabria, nelle zone di Catania, in Sicilia, e a Fondi nel lazio. In sardegna, ad Orgosolo, la lotta ha unito, studenti, operai e pastori. Insieme hanno occupato il municipio e cercato di costruire un potere realmente popolare. Lo scontro con i padroni assume forme sempre più dure e radicali(occupazioni di stazioni, picchetti duri, fabbriche occupate).

Figure 1. A document from the archives of the Gramsci Institute, talking about the FIAT company (Copyright: Fondazione Istituto piemontese Antonio Gramsci onlus).

A valid alternative to fully automatic processing is to involve users in the annotation process, by enabling them to build machine-readable information about document content while exploiting NLP (in particular, Information Extraction) to support them in this activity. This is the perspective we committed to in the PRiSMHA project.

PRiSMHA (Providing Rich Semantic Metadata for Historical Archives) [2] is a three-year (2017–2020) national project, involving the Computer Science and the Historical Studies Departments of the University of Torino. The project is funded by Fondazione Compagnia di San Paolo and Università di Torino, and it is based on the close collaboration with the Polo del '900 (www.polodel900.it, accessed on 10 May 2021), in particular with the Fondazione Istituto piemontese Antonio Gramsci (www.gramscitorino.it, accessed on 10 May 2021), which is the major contributor of the Polo del '900 historical archive (www.polodel900.it/9centro, accessed on 10 May 2021).

The main goal of PRiSMHA is the design and implementation of an ontology-driven prototype platform supporting users in semantic metadata production. The assumption behind the approach adopted in the project, shared within the Digital Humanities community (see, for instance, [3–5]), is that an effective and engaging access to historical archives needs a rich semantic knowledge, based on a domain ontology, describing the content of archival documents.

In order to demonstrate the sustainability of this approach, we focused on the Italian political and social history of the 20th century, taking especially into account the years from the 1960s to the 1990s. Referring to such a domain, we selected 200 documents from the Istituto Gramsci's collections, mainly newspaper/review articles and typewritten leaflets, often with handwritten annotations (see Figure 1). Moreover, we developed an ontology-driven prototype web platform, enabling users to annotate such documents (see Section 2).

However, ontology-based annotation is a very challenging task, also for expert users, especially since it is extremely time-consuming. For this reason, we implemented a double support for the users of the PRiSMHA annotation platform, exploiting both automatic Information Extraction techniques, namely Named Entity Recognition (NER), when full text is available, and entity linking to Linked Open Data (LOD) sets (see [6] for details about this double support). In particular, as regards the support provided by external datasets, the first step has been the integration of a functionality enabling Wikidata [7] search. This functionality provides users with (a) the possibility of linking an entity (i.e., a person, an organization, or a place) belonging to the PRiSMHA knowledge base to a Wikidata entity; (b) the suggestion of values for the *category* and *label* fields in the form the user has to fill in to characterize the new entity (see Section 2).

We evaluated the double support with users, and the results showed that NER and entity linking to LOD actually support users in the annotation activity in an effective way; in particular, users appreciated the pre-filling of *category* and *label* fields in the form for creating a new entity (see [6]). Moreover, in the free comments, some users asked for more effective help, i.e., for suggestions about other properties characterizing the entity in focus. In the same study, some participants provided us with another reason to offer a better support when building semantic metadata: some of them claimed that the task was quite complex, since typically, several entities need to be characterized in order to describe the content of a text fragment. For example, when describing an event (e.g., a protest march) mentioned in a document, users may need to characterize all the single involved entities (such as place, time, people, organizations, etc.), and gathering information about such entities can be time consuming and even distracting from the original task (i.e., the description of the event itself).

On the basis of these suggestions we designed, implemented, and tested a new version of the support system which involved the following when characterizing a new entity (e.g., a person): (a) more properties, besides *category* and *label*, are automatically pre-filled in; (b) more complex semantic representations are suggested, for example the creation of one or more events, in which the entity in focus is involved with a specific role (see Sections 3 and 4 for details). In the proof-of-concept prototype of this new version of the annotation platform, both kinds of suggestions are based on information retrieved from Wikidata.

Two main issues have to be faced in order to reach this goal: (a) the alignment of the semantic model underlying Wikidata with the ontology used in the PRiSMHA system (described in Section 3); (b) on the basis of such an alignment, an effective process to extract useful information from Wikidata and a User Interface to provide users with suggestions (described in Section 4).

With this in mind, we can now formulate the research question we aim at answering with the work presented in this paper:

> *Given an ontology-driven web-based semantic annotation system, can information extracted from available external resources (Linked Open Data), such as Wikidata, provide users with useful suggestions in the creation of new entities used for the annotation?*

The main contribution of this paper is to answer this research question by describing how we aligned the Wikidata semantic model and the PRiSMHA ontology (Section 3) and how this enabled us to extract information that represents the envisioned suggestions (Section 4).

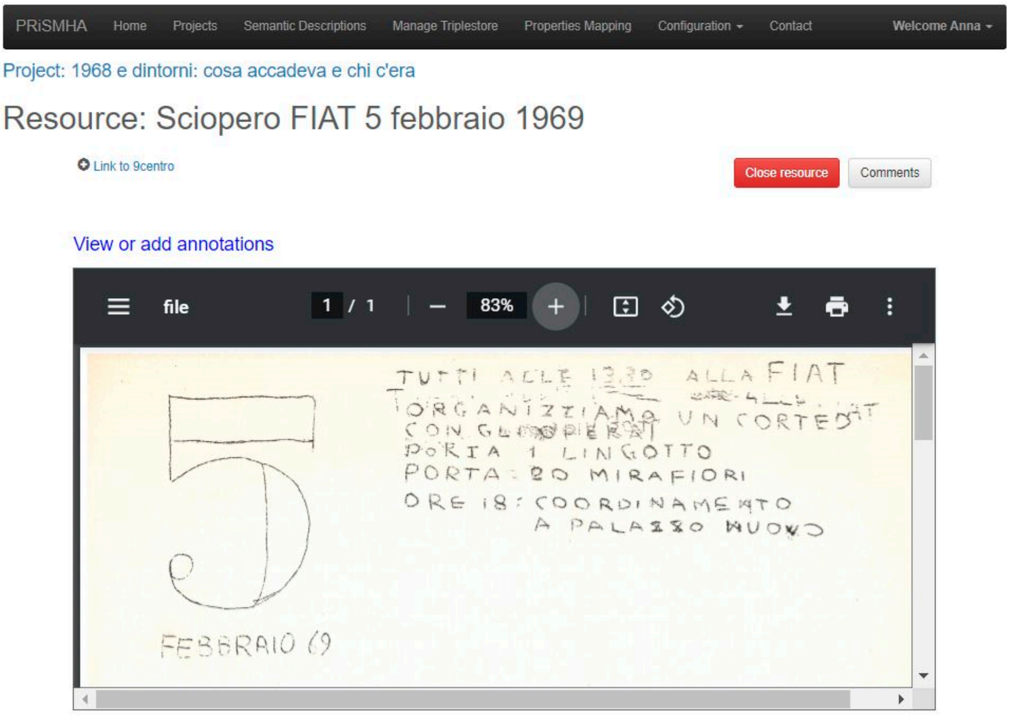
The rest of the paper is organized as follows. In Section 2, we provide an overview of the PRiSMHA environment by presenting the annotation platform and the underlying ontology. Section 3 describes the mappings between the Wikidata semantic model and the PRiSMHA ontology. In particular, Section 3.1 explains how we defined the mapping between Wikidata categories and classes/individuals in the PRiSMHA ontology, while Section 3.1 presents the mappings between Wikidata and PRiSMHA entity description patterns. Section 4 outlines the process used in PRiSMHA to retrieve information from

Wikidata and to provide suggestions to the users who are building semantic representations using the annotation platform. Section 5 presents the results of a qualitative evaluation of the suggestions and discusses them. Finally, Section 6 analyzes different fields where relevant approaches have been developed and discussed, in particular the field of ontology matching, and explains our choices. Section 7 concludes the paper by sketching future work directions.

2. Overview of the PRiSMHA Environment

The core of our project is the ontology-driven prototype platform supporting users in the annotation activity. In PRiSMHA, an "annotation" is the link from a document (more specifically, a document fragment) and a semantic representation stored in a *Semantic KB*. For example, the document in Figure 1 could be annotated with a link to the semantic representation of the entity representing the FIAT company.

In order to add annotations to a selected document, users can click *View or add annotations* (Figure 2): the system pops up a window (Figure 3) where the user can select an existing entity or add a new one by clicking on *Add (and link) new entity*. In the latter case, after having selected the suited entity type (e.g., *Azienda—Company*) and having provided a label (e.g., "FIAT"), the user has to fill in a form (see Figure 4) in which all properties characterizing the entity are listed (actually, properties are organized in three tabs, containing *important*, *useful*, and *other* properties, respectively; see [4]).



Project: 1968 e dintorni: cosa accadeva e chi c'era

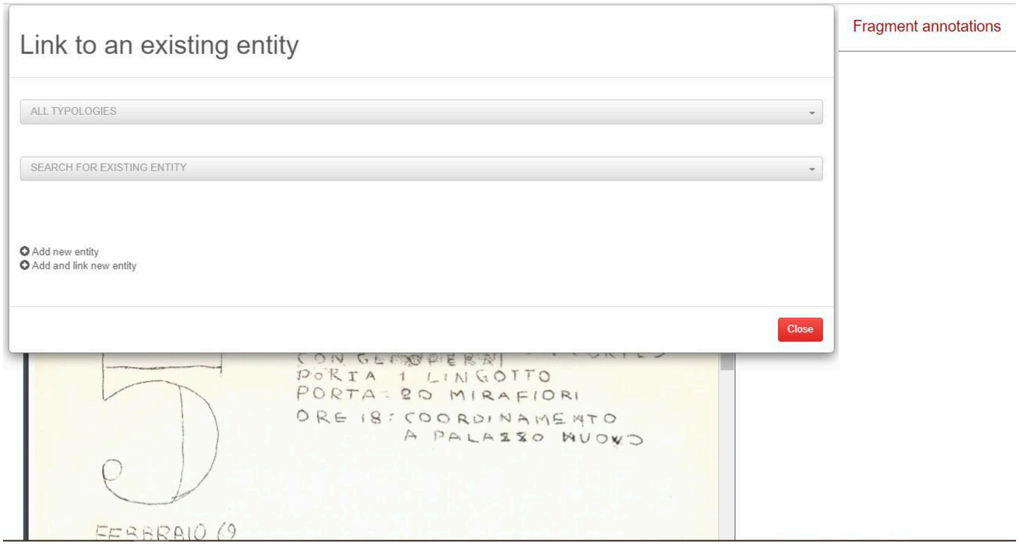
Resource: Sciopero FIAT 5 febbraio 1969

TUTTI ALLE 13.30 ALLA FIAT
ORGANIZZIAMO UN CORTEO
CON GLI OPERAI
PORTA 1 LINGOTTO
PORTA 20 MIRAFIORI
ORE 18: COORDINAMENTO
A PALAZZO NUOVO

5
FEBBRAIO 69

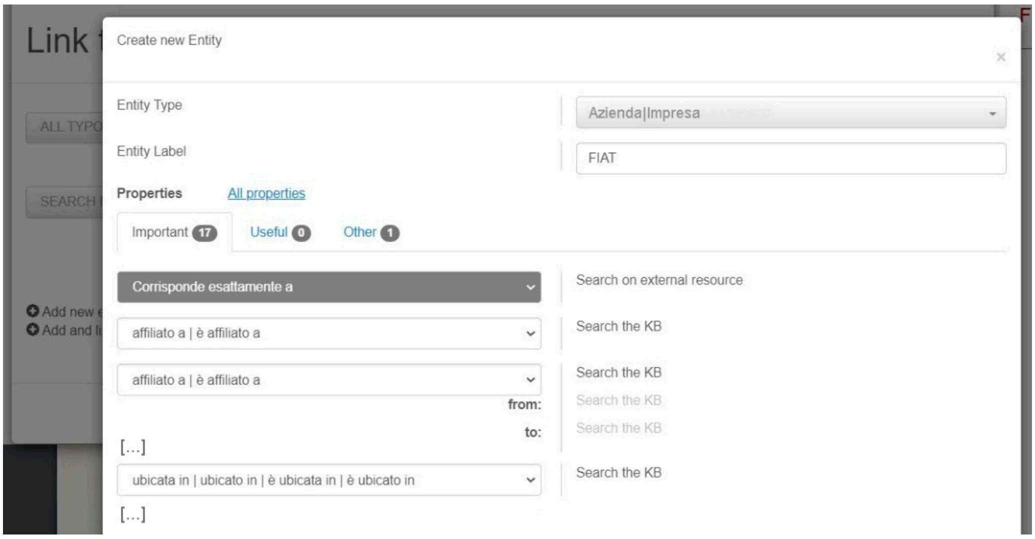
Figure 2. The page for annotating a document on the PRiSMHA prototype platform.

For all the listed properties but the first one, the user is invited to search the KB to select an existing entity (e.g., *Torino* for the *ubicata in—located in* property); if no suitable entity is found, the user has the possibility to create a *basic entity*, i.e., an entity characterized only by type and label. This mechanism has been designed in order to enable the user to go on with the description of the entity in focus (e.g., *FIAT*), without being distracted by the description of property fillers; obviously, she will be able to enhance the description of basic entities later on.



Link to an existing entity

Figure 3. The modal window enabling users to select or add an entity.



Create new Entity

Entity Type	Azienda|Impresa
Entity Label	FIAT

Properties
All properties
Important 17
Useful 0
Other 1
Corrisponde esattamente a
Search on external resource
affiliato a | è affiliato a
Search the KB
affiliato a | è affiliato a
Search the KB
from:
Search the KB
to:
Search the KB
[...]
ubicata in | ubicato in | è ubicata in | è ubicato in
Search the KB
[...]

Figure 4. Part of the form for the characterization of an entity (e.g., a company).

However, as users told us, filling in forms for the characterization of entities to be used in the annotation is a challenging task that is sometimes boring and time consuming. For this reason, in the new version of the prototype we are going to describe in this paper, the user can click *Search on external resource* (corresponding to the filler of the first "property", *Corrisponde esattamente a—Exactly corresponds to*), thus asking the system to search for suggestions provided by external resources, such as Wikidata, in order to pre-fill in some of the properties.

In Sections 3 and 4, we will describe in detail how information is retrieved and suggestions are generated. Here, we briefly present the main characteristic of the annotation platform, i.e., the fact that it is ontology-driven.

The system's Semantic KB stores the formal descriptions of the entities used in the annotation of documents, as well as the annotation themselves (i.e., the links between entities and documents). It is implemented (by means of Jena TDB 3.6.0 (jena.apache.org/documentation/td as an RDF triplestore (www.w3.org/RDF, accessed on 10 May 2021)).

The conceptual vocabulary used to describe entities is represented by two ontologies, namely HERO and HERO-900.

HERO (Historical Event Representation Ontology; available at w3id.org/hero/HERO, accessed on 10 May 2021) is a core ontology organized into five modules:

- HERO-TOP is the upper level of the ontological model, and it is based on the basic distinctions defined in DOLCE [8]: *perdurants* (including *states* and *events*), *objects* (including *physical objects* and *non-physical objects*, such as *social objects* as *organizations* or *social roles*), and *abstract entities*. HERO-TOP also contains relations such as, for example, the *participation* relation linking *objects* (e.g., people) to *perdurants* (e.g., *events* they participate in).
- HERO-ROCS is the module devoted to the representation of *social roles* (e.g., Prime Minister), *organizations* (e.g., Italian Communist Party, FIAT), and *collective entities* (e.g., politicians). Since *social roles* are particularly relevant in the mechanism for suggesting information extracted from Wikidata, we will describe below how they are represented in HERO.
- HERO-EVENTS is the module devoted to the representation of *events* (e.g., homicide). Events, together with their participants, play a major role in suggestions of information extracted from Wikidata, so some details of this module will be described below.
- HERO-PLACE is the module devoted to the representation of *places* (e.g., cities).
- HERO-TIME is the module devoted to the representation of *time intervals* (e.g., days and years).

HERO-900 is a domain ontology composed by three modules (HERO-ROCS-900, HERO-EVENT-900, and HERO-PLACE-900) that extend the corresponding HERO modules with the definition of classes and relations characterizing the history of the 20th century, with a special focus on the years from the 1960s to the 1990s in Italy.

The version of HERO + HERO-900 used in the current prototype is encoded in OWL 2 (www.w3.org/OWL, accessed on 10 May 2021) and counts 429 classes, 380 properties, 145 individuals, and 4661 logical axioms.

In the following, we provide some details about how HERO and HERO-900 model the semantic representation of an entity, focusing on those aspects that are particularly relevant to the suggestion mechanism that we will describe in Sections 3 and 4.

Simple properties. In HERO, there are *data properties* and *object properties*. An example of data property is *hasName*, which can be used to assign a name, in the form of a string, to an entity; for example, the triple <*SandroPertini*, *hasName*, "Sandro Pertini"> states that the name of *SandroPertini* (an instance of the *PhysicalPerson* HERO class) is "Sandro Pertini". An example of object property is *hasBirthPlace*, which can be used to say that somebody was born somewhere; for example, the triple <*SandroPertini*, *hasBirthPlace*, *StellaSanGiovanni*> states that *SandroPertini* was born in *StellaSanGiovanni* (an instance of the *ItalianMunicipality* HERO class).

When the user is filling in the form to characterize a new entity to be added to the Semantic KB, an algorithm (described in [4]) singles out the suitable properties for that entity, as well as the valid classes for their fillers.

Temporary properties. In some cases, the attribution of a property to an entity has to be limited to a specific time interval; for example, a politician can be affiliated to a given political party only for a specific period of time. In such cases, a property attribution involves at least three arguments: the two entities at issue and the temporal parameter. Since n-ary (n > 2) properties cannot be directly represented in OWL 2, HERO models them by means of specific classes whose instances represent particular property attributions (www.w3.org/TR/swbp-n-aryRelations, accessed on 10 May 2021). In the following, we describe how temporary properties are modeled in HERO, taking the *temporary affiliation* as an example. To state that Sergio Garavini has been affiliated to the Italian Communist Refoundation Party from 1991 to 1995, we create an instance *x* of the HERO *TemporaryAffiliationToOrganization* class, and assert the following triples:

- <*x*, *hasAffiliationToOrganizationEntityElement*, *SergioGaravini*>, stating that Sergio Garavini has been temporary affiliated to some organization (where *SergioGaravini* is an instance of the class *PhysicalPerson*);

- *<x, hasAffiliationToOrganizationOrganizationElement, ItalianCommunistRefoundationParty>*, stating that *ItalianCommunistRefoundationParty* is the object of the temporary affiliation (where *ItalianCommunistRefoundationParty* is an instance of the *PoliticalParty* class, subclass of *Organization*);
- *<x, hasTemporalParameterElement, tx>*, where *tx* is an instance of the *TimeInterval* class, used to define the start and end time boundaries of the affiliation;
- *<tx, intBeginsIn, t1991>*, stating that *t1991* (instance of the *Year* class, subclass of *TimeInterval*) is the start time of the affiliation;
- *<tx, intEndsIn, t1995>*, stating that *t1995* (instance of the *Year* class, subclass of *TimeInterval*) is the end time of the affiliation.

Playing roles. HERO enables users to state that somebody played a specific social role, for example, that Aldo Moro was professor at Sapienza University of Rome.

Following a well-known approach [9], we reify social roles by creating instances of the *Role* class (or of a subclass of it); then, instances representing roles can be "institutionalized" by an organization [10] and "played" by a person: For example, the role "professor at Sapienza University of Rome" can be institutionalized by the Sapienza University and played by Aldo Moro.

Slightly more formally, to model the mentioned example, we can use *professor*, instance of the *Profession* class (subclass of *Role*) already available in HERO (where a number of individuals representing common social roles within the domain are present); moreover, a new instance of the *Profession* class (subclass of *Role*), *professorAtSapienza*, can be created, and the following triples can be stated:

- *<professorAtSapienza, subRoleOf, professor>*, stating that the *professorAtSapienza* role is a sub-role of the *professor* role;
- *<Sapienza, institutionalizes, professorAtSapienza>*, stating that the *professorAtSapienza* role is institutionalized by the Sapienza University (where *Sapienza* is an instance of *University*, subclass of *Organization*);
- *<AldoMoro, isClassifiesdBy, professorAtSapienza>*, stating that *AldoMoro* (instance of *PhysicalPerson*) played the *professorAtSapienza* role; the *isClassifiedBy* property expresses the main relation between concepts and ground entities; when the concept is a role, it expresses the notion of playing such a role.

If information about time is available, the temporary version of *isClassifiedBy* ("plays role") can be used. The pattern is the same as that of temporary affiliation described above: An instance *x* of the *TemporaryClassification* class is created, and the *hasClassificationClassifiedEntityElement* and *hasClassificationClassifyingConceptElement* properties are used to state the person and the role involved, respectively; time boundaries are represented as in *temporary affiliation*.

Participating in events with a specific role. HERO enables users to represent events with their participants, as well as the specific roles they play in the events. Inspired by the neo-Davidsonian approach to the representation of events [11], participants and their roles are defined by means of binary properties (corresponding to *thematic roles*) [12].

Consider the following example: Giovanni Falcone was assassinated by Giovanni Brusca on May 23 1992 in Capaci; an instance 'e' of the *Homicide* class is created and the following triples are stated:

- *<e, hasPatient, GiovanniFalcone>*, stating that *GiovanniFalcone* (instance of *PhysicalPerson*) "participated" in the homicide e with the role of "patient" (i.e., the participant affected by the event);
- *<e, hasAgent, GiovanniBrusca>*, stating that *GiovanniBrusca* (instance of *PhysicalPerson*) participated in the homicide e with the role of "agent" (i.e., the participant who voluntarily acted in the event);
- *<e, hasLocation, Capaci>*, stating that the event e took place in *Capaci* (instance of *ItalianMunicipality*);

- *<e, hasTimespan, 23-05-1992>*, stating that the event e took place on May 23 1992 (*23-05-1992*, instance of the *Day* class, subclass of *TimeInterval*).

In Section 4, we will describe how we extract information from Wikidata to provide suggestions to the users of the annotation platform, but the first step needed to design such a mechanism is the alignment of the semantic model underlying Wikidata with HERO (Section 3). In the rest of the paper, for the sake of readability, we will refer to the two ontologies HERO and HERO-900 simply as HERO.

3. Mapping from the Wikidata Semantic Model to HERO

As stated before, the first step toward the interoperability between Wikidata and the PRiSMHA platform is the definition of an alignment between the two semantic models. In particular, we aim at defining a directed alignment, i.e., a *mapping* (see Section 6), in which the Wikidata model represents the *source ontology* and HERO the *target ontology* [13].

Since Wikidata is a cross-domain semantic model, while HERO, and in particular HERO-900, is a specific domain ontology, the alignment process is driven by HERO: We search the Wikidata model to find correspondences with a given HERO concept.

As far as HERO classes and individuals are concerned, we designed and implemented an automatic algorithm (described in Section 3.1) that provided us with a set of correspondences between the two ontologies; then, this set has been manually checked and corrected in order to produce the final mappings used by the system to provide suggestions.

As far as properties are concerned, the results of the automatic algorithm were definitely unsatisfactory. Moreover, as we will show in Section 3.2, in many cases, Wikidata complex patterns involving properties have to be mapped onto complex patterns in HERO. For these reasons, such mappings have been manually defined.

We recall our goal, i.e., providing users with suggestions based on information retrieved by Wikidata: In this perspective, the matching task is not our goal, but it is a needed step aimed at enabling suggestions. In Section 6, we will discuss how our work to align Wikidata and HERO semantic models is related to existing ontology matching approaches.

3.1. Mappings between Categories

In this section, we describe the approach we developed to find the Wikidata categories, which correspond to HERO classes or individuals. The strategy that we devised is based on label matching. Since HERO is provided with both Italian and English labels, the matching process relies on both languages. The label-matching strategy consists of a sequence of steps, starting with a class or individual from the HERO ontology to end up with the category in Wikidata which has the same meaning as the input class. In particular, for each HERO class or individual, we perform: (1) Label Extraction, (2) Knowledge Base Access, and (3) Scoring.

In Section 6, we will discuss ontology matching approaches and why we decided to define a new algorithm instead of using existing tools.

(1) Label Extraction.

The Label Extraction step is aimed at extracting the labels for English and Italian for the HERO category provided as input; then, the extracted lexicalizations are used to access external knowledge bases in the second step. The intuition underlying the labels extraction step is to exploit both languages to improve the precision of the matching strategy. The extraction process for the HERO category (either a class or an individual) C produces the two sets ${L^C}_{en}$ and ${L^C}_{it}$ of the English and Italian labels, respectively, which are associated with C. In order to increase the precision of the matching strategy, we build the Cartesian product $L^C = {L^C}_{en} \times {L^C}_{it}$ containing all the associations of each English label with each Italian label for the HERO category C.

We introduce an example to illustrate the strategy. A graphical representation of the whole process is depicted in Figure 5. Let us consider the HERO *Questioning* class. In HERO, *Questioning* is provided with the two sets of labels: ${L^{Questioning}}_{en}$ = *{questioning, interrogation}* and $L^{Questioning}{}_{it}$ = *{interrogatorio}* (where *interrogatorio* is the Italian translation of *questioning*). Thus, the Cartesian product $L^{Questioning}$ = *{<interrogation, interrogatorio>,<questioning, interrogatorio>}* is the output of this step.

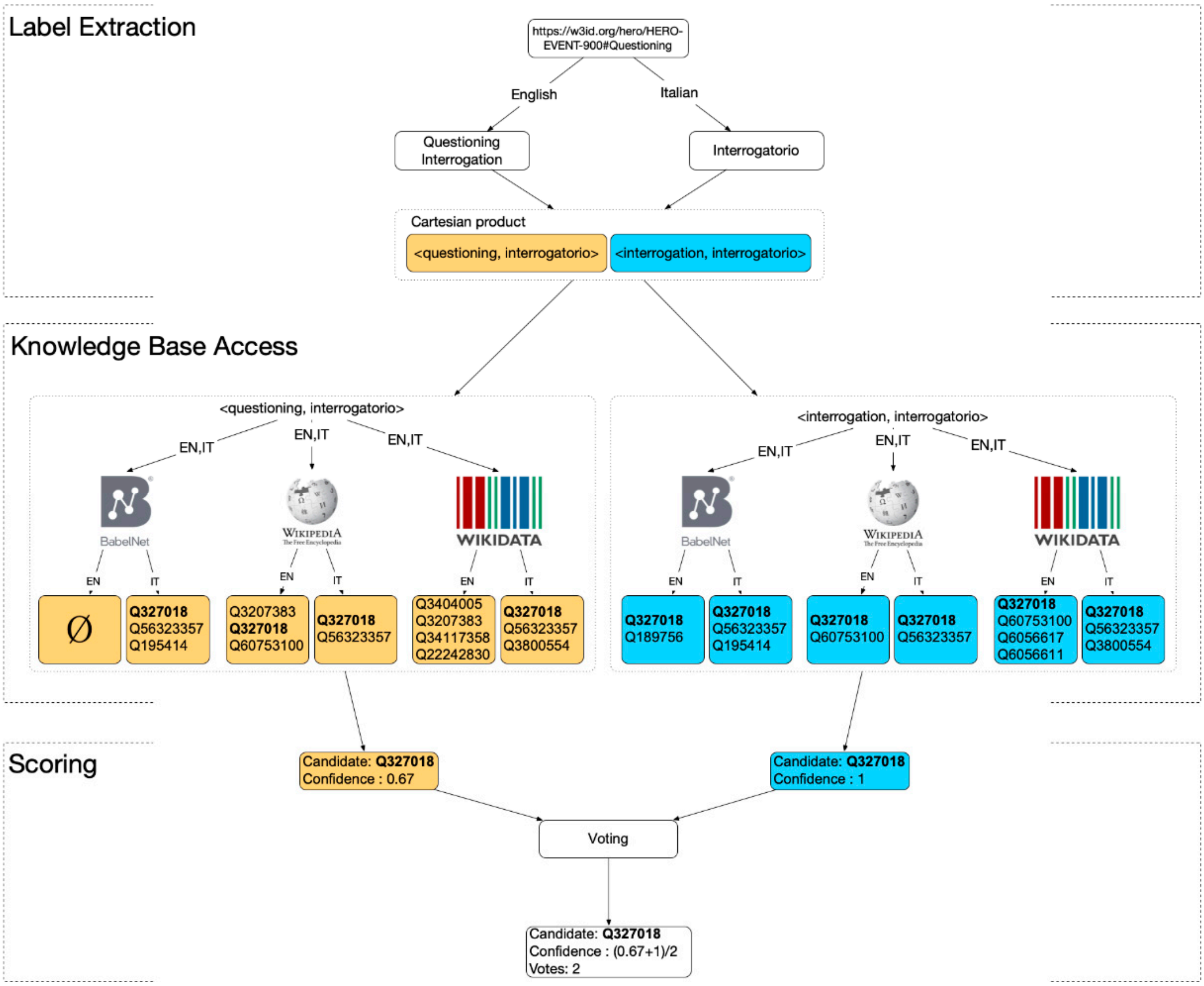


Figure 5. Picture representing the automatic matching strategy. The three steps, namely Label Extraction, Knowledge Base Access, and Scoring are depicted by starting from the HERO *Questioning* label.

(2) Knowledge Base Access.

The Knowledge Base Access step is aimed at retrieving the best matching entity for each pair of labels in L^C by querying external knowledge bases. Since we are interested in providing matches grounded in the Wikidata space, we exploit two additional resources for which a mapping to Wikidata exists (queries to Wikidata have been performed through the MediaWiki API: www.wikidata.org/w/api.php, accessed on 23 March 2021): BabelNet [14] (queries to BabelNet have been performed through the Java API: babelnet.org/guide, accessed on 23 March 2021) and Wikipedia (en.wikipedia.org, accessed on 23 March 2021; queries to Wikipedia have been performed through the MediaWiki API: en.wikipedia.org/w/api.php, accessed on 23 March 2021).

The rationale underlying the decision to leverage multiple resources instead of using Wikidata only is that by performing the same query on multiple resources that share the same vocabulary, we can, in principle, obtain different rankings for the same entities, according to the different sorting criteria implemented by different resources. Additionally, if the same entity is retrieved from multiple resources by using the same labels, we can reasonably consider the result as more reliable: the more resources provide the same entity by means of the same query, the more reliable should be the result.

Since we are interested in computing the best matching entity for each label pair <$l^C{}_{en}$, $l^C{}_{it}$> contained in L^C, we perform six different queries, one for each resource and language, in order to obtain six different result rankings. We can represent the query to a knowledge base with the function *Q(KB, label)*, where *KB* is the resource and *label* is the English or Italian label from L^C. For example, the query to Wikipedia with the label *interrogation* can be represented as *Q(Wikipedia, interrogation)*.

Let us consider the HERO *Questioning* class, for which the Label Extraction step returns the set $L^{Questioning}$ *= {<interrogation, interrogatorio>,<questioning, interrogatorio>}*. If we consider the first pair of labels <*interrogation, interrogatorio*> as input for the Knowledge Base Access step, we perform six different queries, three for the English labels and three for the Italian labels, obtaining the following results (as regards Wikipedia, for the sake of brevity, we only report the first two results, referred to with the title of the corresponding Wikipedia pages; all the query results are ordered lists):

- *Q(Wikipedia, interrogation) = [Interrogation, Interrogation_(TV_series)];*
- *Q(Wikipedia, interrogatorio) = [Interrogatorio, Interrogatorio_(ordinamento_italiano)];*
- *Q(Wikidata, interrogation) = [Q327018, Q60753100, Q6056617, Q6056611];*
- *Q(Wikidata, interrogatorio) = [Q327018, Q56323357, Q3800554];*
- *Q(BabelNet, interrogation) = [bn:00032094n, bn:00047209n, bn:00047208n];*
- *Q(BabelNet, interrogatorio) = [bn:00032094n, bn:22438476n, bn:00023955n].*

The query to a knowledge base KB provides results that lie in the space S^{KB} of entities belonging to the queried knowledge base. For example, *Q(Wikipedia, interrogation)* returns an ordered list whose elements are Wikipedia entities. Since we are interested in obtaining entities in the Wikidata space, we defined a mapping function *M* that maps each Wikipedia or BabelNet list of entities to the corresponding Wikidata ones: M:$\mathrm{S}^{\mathrm{Wikipedia}}$ ∪ $\mathrm{S}^{\mathrm{BabelNet}} \rightarrow \mathrm{S}^{\mathrm{Wikidata}}$. In practice, *M* exploits the equivalence links to Wikidata entities possibly provided by Wikipedia and BabelNet: Wikipedia pages offer such links in the section titled "In other projects", while BabelNet synsets provide them in their sources set. All those Wikipedia or BabelNet entities for which no link to a Wikidata entry exists are lost in the mapping. For example, given the list of Wikipedia entities *[Interrogation, Interrogation_(TV_series)]*, we have *M([Interrogation, Interrogation_(TV_series)]) = [Q327018,Q60753100]*; i.e., the corresponding list of Wikidata entities.

If we apply the mapping function *M* to the query results listed above, we obtain the following entity lists in the Wikidata space:

- *M(Q(Wikipedia, interrogation)) = [Q327018, Q60753100];*
- *M(Q(Wikipedia, interrogatorio)) = [Q327018, Q56323357];*
- *M(Q(Wikidata, interrogation)) = [Q327018, Q60753100, Q6056617, Q6056611];*
- *M(Q(Wikidata, interrogatorio)) = [Q327018, Q56323357, Q3800554];*
- *M(Q(BabelNet, interrogation)) = [Q327018, Q189756]* (in this case, the mapping function returns only two correspondences, since the *bn:00047209n BabelNet* synset does not have any Wikidata correspondence);
- *M(Q(BabelNet, interrogatorio)) = [Q327018, Q56323357, Q195414].*

Therefore, the outputs of the Knowledge Base Access step are six ordered lists of elements in the Wikidata space, for each <$l^C{}_{en}$, $l^C{}_{it}$> ∈ L^C. In our example, we obtain the six lists for the pair of labels <*interrogation, interrogatorio*>, as well as the six lists for <*questioning, interrogatorio*> (see Figure 5).

(3) **Scoring.** The Scoring step is composed of the Candidates Selection and Voting substeps.

(3.1) **Candidates Selection.** On the basis of the results provided by the mapping function *M*, we compute the best candidate for the pair of labels <$l^C{}_{en}$, $l^C{}_{it}$>. To this aim, we take into consideration the set $Candidates_{<lCen,\ lCit>}$ of candidates obtained by intersecting the six resulting lists (note that this intersection may be empty, since a query may provide no result and the mapping of a query result may return an empty set; in this case, we consider the maximum number of mapped query results that provide a non-empty intersection; if all the mapped query results are empty, no candidate is returned for the considered pair of labels). In our example, we have $Candidates_{<interrogation,\ interrogatorio>}$ = *{Q327018}*, and computing the best candidate is trivial. In general, $Candidates_{<lCen,\ lCit>}$ may contain more than one element. In that case, we score all candidates to single out the best one. To this purpose, we define the function $S(wdc_i) = \sum_{L \in LS} position(wdc_i, L)$, which computes the score of the candidate $wdc_i \in Candidates_{<lCen,\ lCit>}$, where $position(wdc_i, L)$ returns the position of wdc_i in the list *L* belonging to the set *LS* of the intersected lists of results. In our example, *S(Q327018)* = 1 + 1 + 1 + 1 + 1 + 1 = 6, because *Q327018* always appears as the first element in each list of results. Since the scoring function sums the ranking positions, lower scores correspond to candidates that, on average, are in a higher position in the resulting rankings. Thus, we select the candidate in $Candidates_{<lCen,\ lCit>}$ with the minimum score: $bestCandidate_{<lCen,\ lCit>} = argmin_{wdci \in Candidates<lCen,\ lCit>}\ S(wdc_i)$. Moreover, a confidence value $Confidence_{<lCen,\ lCit>}(bestCandidate_{<lCen,\ lCit>})$ is associated with the best candidate, defined as the number of the lists of results that contain the best candidate, divided by the total number of queries. In our example, $Confidence_{<interrogation,interrogatorio>}(Q327018) = 1$, because *Q327018* occurs in each list of results. Therefore, the output of the Candidates Selection substep for the HERO category *C* is the set $Candidates^C$ containing a pair < $bestCandidate_{<lCen,\ lCit>}$, $Confidence_{<lCen,\ lCit>}(bestCandidate_{<lCen,\ lCit>})$> for each <$l^C{}_{en}$, $l^C{}_{it}$> ∈ L^C for which the Knowledge Base Access step returned at least one non-empty list. In our example, $Candidates^{Questioning}$ = *{<Q327018, 1>,<Q327018, 0.67>}* (Figure 5).

(3.2) **Voting.** The Voting substep either returns the Wikidata category corresponding to the HERO category C or it answers that no correspondence exists. Basically, it selects in $Candidates^C$ the best matching Wikidata category *BM(C)* for the HERO category *C*. It is worth noting that $Candidates^C$ may contain multiple occurrences for a same Wikidata category (e.g., in our example, *Q327018* occurs twice). We adopt a major voting strategy and select the candidate with the maximum number of occurrences as the best matching Wikidata category for the HERO category *C*, i.e., $M(C) = argmax_{wdci \in Cat(CandidatesC)}\ Count(wdc_i, Candidates^C)$, where $Cat(Candidates^C)$ is the set of Wikidata categories occurring in $Candidates^C$ and $Count(wdc_i, Candidates^C)$ counts the occurrences of wdc_i in $Candidates^C$. As confidence score for the best matching Wikidata category, we take the arithmetic mean of the confidence scores associated with its occurrences in $Candidates^C$: If this confidence score is greater than an established threshold θ, *BM(C)* is returned as the Wikidata category corresponding to the HERO category *C*; otherwise, no correspondence is returned for *C*. In our example, the Candidates Selection substep returns a set of candidates containing only two occurrences of the *Q327018* Wikidata category; then, *BM(Questioning)* = *Q327018*, with confidence 0.835. Since in our case, we set the threshold θ = *0.15*, *Q327018* is returned as the Wikidata category corresponding to the *Questioning* HERO class. Whenever the list $Candidates^C$ contains several candidates with the same greatest number of votes, we select the one with the greatest confidence. If there is a tie on the confidence score too, we choose the best ranked candidate obtained by querying Wikidata using the English language.

If none of them occurs in the results of that query, we randomly choose one of them. The whole process does not always return a correspondence for the HERO category *C*. In fact, the following may happen:

- The Candidates Selection substep returns $Candidates^C = \emptyset$. This may happen when the Knowledge Base Access step returns only empty lists for each pair of labels $<l^C{}_{en}, l^C{}_{it}> \in L^C$;
- The Voting substep finds a best matching Wikidata category *M(C)* with a confidence score lower than the threshold θ.

Evaluation of the approach.

We exploited the same approach for both classes and individuals in HERO. The ontology contains 429 classes; 189 of them have a corresponding entity in the Wikidata space, while for 240, no correspondence exists. Additionally, the ontology contains 145 individuals: For the vast majority of them, a correspondence does exist, and for only 32 of them, it does not. Then, we manually fixed the matching strategy errors by either choosing the appropriate Wikidata category for HERO categories with a possible match or by setting "no correspondence" for HERO categories with no Wikidata correspondence.

We evaluated our automatic matching strategy in order to estimate its reliability and reusability: Since both Wikidata and HERO are constantly evolving, the same approach will be adopted with new versions of the two models; therefore, we are interested in assessing its suitability in the perspective of its reuse.

We considered the following as correct results: (i) the returned associations between a HERO category and its actual Wikidata corresponding category; (ii) the returned "no correspondence" answer for those HERO categories for which no Wikidata corresponding category exists. We considered the following as wrong results: (i) the returned associations between a HERO category and a wrong Wikidata category, for those HERO categories for which a correspondence exists; (ii) the returned "no correspondence" answer for the HERO categories for which a Wikidata corresponding category does exist; (iii) the returned associations between a HERO category and a Wikidata category, for those HERO categories for which no correspondence exists.

We consider the reference matching set RM of the actual correspondences between HERO and Wikidata categories and the set SM of the correspondences returned by the system. Then, we computed the precision $P = \frac{|\, RM \cap SM \,|}{|SM|}$, the recall $R = \frac{|\, RM \cap SM \,|}{|RM|}$, and the F1-measure (i.e., the harmonic mean of *P* and *R*).

The results are presented in Table 1: Our approach obtains satisfying performances with better results on classes.

Table 1. Precision, recall, and F1-measure of the automatic matching strategy.

	P	*R*	**F1-Measure**
Classes	0.88	0.82	0.85
Individuals	0.77	0.77	0.77

If we consider the results of the evaluation, the false positives are mainly formed by the cases for which there exists no correspondence in the Wikidata domain, nevertheless, the algorithm provides a candidate: Such errors sum up to 52 out of 75 for classes and 20 out of 35 for individuals. Since this part of false positives is the most incisive, we further analyzed such errors. We noticed that the vast majority of them are provided with a low confidence score, i.e., 0.33 and 0.17. Additionally, such errors seem to be driven by some specific circumstances: (a) very specific and localized concepts, such as HERO *trade union secretariat* or *temporary layoff*; (b) specific actions belonging to our domain, such as HERO *wounding*, which is intended as the act of inflicting a wound, or HERO *police charge*, which is intended as the act of charging by a police formation. For these actions, the label of the concept in HERO consists of multiple words, and at least one of them has a general meaning, which is fully represented in general purpose knowledge bases (e.g., layoff, police, trade union). In such cases, the query returns the general concept, hiding the lack of the specific one. For example, the query for "police charge" returns the Wikidata category *police* (Q35535) instead of the empty set.

If checking the correctness of a returned correspondence is rather easy, checking the correctness of a "no correspondence" answer requires browsing Wikidata to look for a possible match. For this reason, we evaluated the reliability of our approach in stating that no Wikidata correspondence exists for a HERO category: Given the set *CN* of HERO categories for which no correspondence exists, and the set *N* of HERO categories for which the matching systems answers "no correspondence", we estimated the probability of a lack of match for a HERO category, given that the matching system answers"no correspondence" for it, as $P(CN|\ N) = \frac{|CN \cap N|}{|N|}$. The results presented in Table 2 show that the system is highly reliable in this task; therefore, in such cases, we can reasonably save us the effort of checking the system's answers.

Table 2. Estimated probability that a "no correspondence" answer is correct.

	Classes	**Individuals**
P(CN ∣ N)	0.94	1.0

3.2. Mappings between Properties and Mappings Between Complex Patterns

In this section, we present the mappings between Wikidata and HERO patterns involving properties. In some cases, the mapping is straightforward, since the correspondence is defined between two properties. In other cases (actually, the most interesting ones), a Wikidata property, or a Wikidata pattern, corresponds to a complex representation in terms of HERO.

As mentioned before, we tried to use the algorithm developed to map categories (Section 3.1) to obtain mappings between Wikidata and HERO properties, but the results were not so encouraging. Considering the 380 properties defined in HERO, only 57 of them were mapped by the algorithm to a Wikidata property and, analyzing these mappings, we observed that only 15 of them were correct. As a consequence of these results, we analyzed the structure of the Wikidata semantic model, to individuate groups of properties that are noteworthy for our domain.

A first property that needs to be mapped is the Wikidata *instance of* (P31) property, which relates individuals with the categories to which they belong. As reported in Table 3, *instance of* (P31) is mapped to *rdf:type* (mapping 1). Such a mapping states that any Wikidata triple <*?X, instance of, (P31) ?Y*> can be translated in HERO as <*?X, rdf:type, Map-c(?Y)*>, where *Map-c(?Y)* is the HERO category corresponding to the Wikidata category *?Y*, according to the mappings between categories (Section 3.1), or one of its super-categories (actually, the HERO triple should formally be <*Map-i(?X), rdf:type, Map-c(?Y)*>, where *Map-i(?X)* represents the individual in the PRiSMHA triplestore corresponding to the Wikidata individual *?X*. For the sake of simplicity, in this section, we assume that Wikidata and PRiSMHA can formally share instances). The function *Map-c* is computed as follows:

Table 3. Four sample mappings from Wikidata description patterns to HERO ones. Variables are prefixed with the question mark (e.g., *?X*). Variables occurring in Wikidata patterns are intended as universally quantified, while those occurring only in HERO patterns are intended as existentially quantified. For the sake of readability, the latter are named *?H1, . . . ?H4*. *Map-c(?Y)* is the result of the mapping of the Wikidata category *?Y* to a HERO category; in mapping 3, *quals* indicates Wikidata qualifiers, and *conds* expresses conditions for the associated optional triple to be created.

Mapping #	Wikidata Pattern	HERO Pattern
1	<?X instance of(P31) ?Y >	<?X rdf:type Map-c(?Y)>
2	<?X place of birth(P19) ?Y>	<?X hasBirthPlace ?Y>
3	<?X employer(P 108) ?Y> + quals: {<start time(P580) ?T1>, <end time(P 582) ?T2>, <position held(P39) ?Z>}	<?X isAffiliatedTo ?Y> <?H1 rdf:type Profession> <?X isClassifiedBy ?H1> <?Y institutionalizes ?H1> [<?H1 subRoleOf Map-c (?Z)>] conds:{Map-c(?Z) a Role} <?H2 rdf:type TemporaryAffiliationToOrganization> <?H2 hasAffiliationToOrganizationEntityElement ?X> <?H2 hasAffiliationToOrganizationOrganizationElement ?Y> <?H2 hasTemporalParameterElement ?H3> <?H4 rdf:type TemporaryClassification> <?H4 hasClassificationClassifiedEntityElement ?X> <?H4 hasClassificationClassifyingConceptElement ?H1> <?H4 hasTemporalParameterElement ?H3> <?H3 rdf:type TimeInterval> <?H3 intBeginsIn ?T1> <?H3 intEndsIn ?T2>
4	<?X killed by(P157) ?Y>	<?H1 rdf:type Homicide> <?H1 hasPatient ?X> <?H1 hasAgent ?Y > <?H2 rdf:type PersonDeath> <?H2 hasPatient ?X > <?H1 isCausalFactorOf ?H2>

```
Map-c:
input: a Wikidata category WdC;
output: a HERO category equivalent to WdC or to a super-category of WdC
    1. Add WdC to a FIFO queue Q
    2. For each C in Q:
      a. If a correspondence is defined for C,
         return the HERO category specified by the mapping
      b. Add the parents of C to Q
      3. return HEROEntity//if WdC and all its ancestors do not map to any HERO cate-gory,
                              //the most general HERO category is returned.
```

In order to get the parents of a Wikidata category (*C*), we use the following SPARQL query, which retrieves all the fillers (objects) for the Wikidata properties *instance of* (P31) and *subclass of* (P279), in statements where the subject is the Wikidata category *C*.

```
select * where {
  { select ?c where {
        C wdt:P31 ?c.
  }}
    union {
        select ?c where {
    C wdt:P279 ?c.
    }
  }
}
```

Consider, as an example, the Wikidata statement <*Sandro Pertini, instance of (P31), human (Q5)*>. According to mapping 1 in Table 3, this triple is translated in HERO as <*SandroPertini, rdf:type, Map-c(human (Q5))*>. Since no direct correspondence (in the sense specified in Section 3.1) is defined for *human* (Q5), *Map-c* enqueues all its parents, namely: the Wikidata categories *person* (Q215627), *organisms known by a particular common name* (Q55983715), *natural person* (Q154954), and *omnivore* (Q164509), as shown in Figure 6. *Map-c* now considers *person* (Q215627), for which a mapping is defined to the *PhysicalPerson* HERO class. So, *Map-c* returns *PhysicalPerson*; i.e., *Map-c(human (Q5)) = PhysicalPerson*.

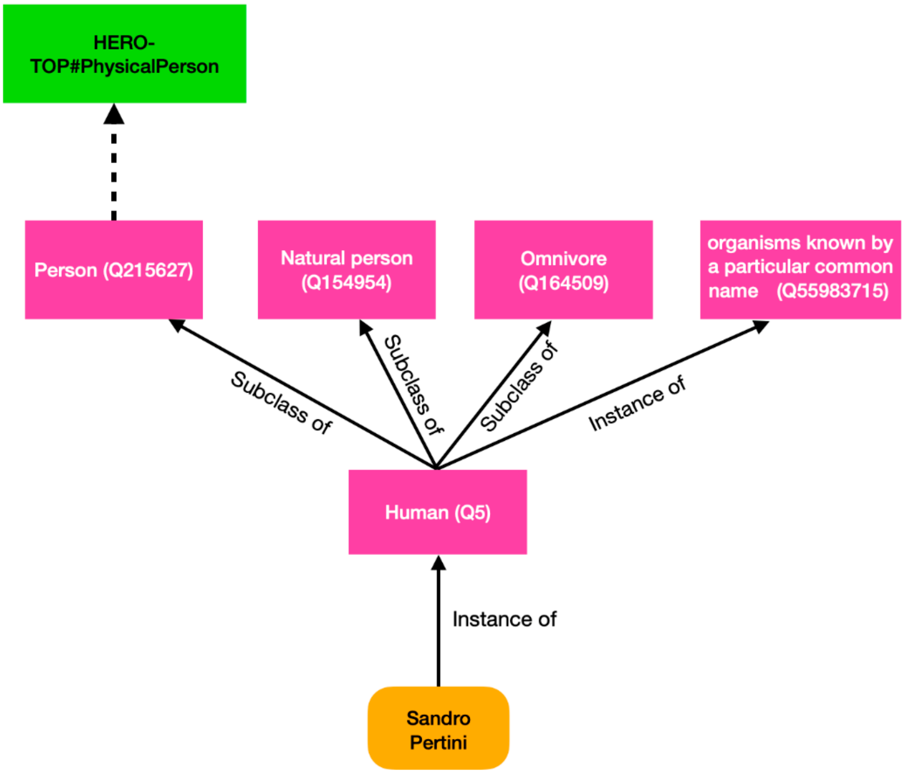
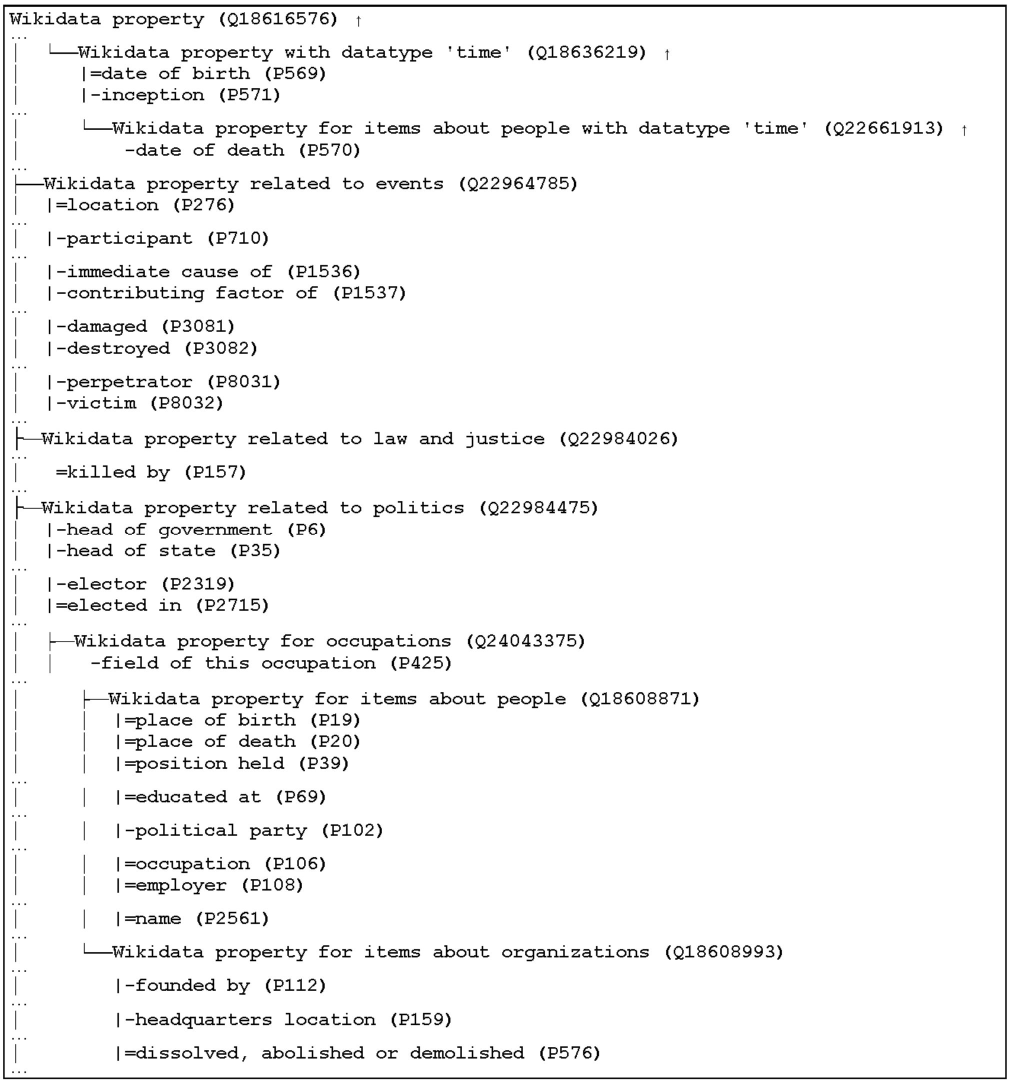


Figure 6. Partial representation of the parents of the *human* (Q5) Wikidata category. Orange boxes represent Wikidata entities, pink boxes represent Wikidata categories, and green boxes represent HERO classes. Dashed arrows represent mappings between Wikidata and HERO categories.

For the analysis of the Wikidata properties other than *instance of* (P31), we used the *Wikidata-Taxonomy* tool (www.npmjs.com/package/wikidata-taxonomy, accessed on 10 May 2021) to retrieve all the subcategories of the category *Wikidata Property* (Q18616576). We analyzed the extracted taxonomy and we considered the property categories that are more relevant for the PRiSMHA domain, i.e., those related to people's life, history of organizations, social roles, participation in events, etc. Then, for the selected categories, we extracted all the corresponding instances. Figure 7 shows a "collage" of the extracted properties.



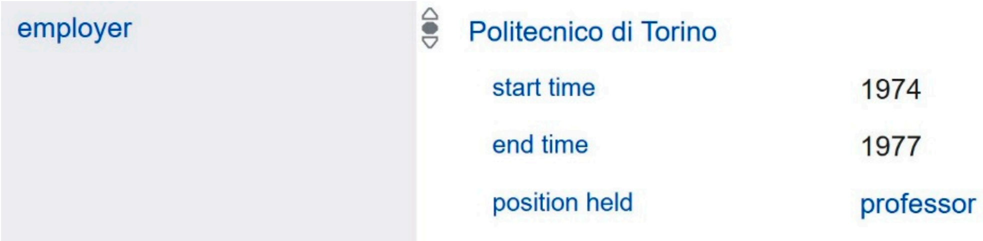
```
Wikidata property (Q18616576) ↑
...
│  └──Wikidata property with datatype 'time' (Q18636219) ↑
│     |=date of birth (P569)
│     |-inception (P571)
...
│     └──Wikidata property for items about people with datatype 'time' (Q22661913) ↑
│        -date of death (P570)
...
├──Wikidata property related to events (Q22964785)
│  |=location (P276)
...
│  |-participant (P710)
...
│  |-immediate cause of (P1536)
│  |-contributing factor of (P1537)
...
│  |-damaged (P3081)
│  |-destroyed (P3082)
...
│  |-perpetrator (P8031)
│  |-victim (P8032)
...
├──Wikidata property related to law and justice (Q22984026)
...
│   =killed by (P157)
...
├──Wikidata property related to politics (Q22984475)
│  |-head of government (P6)
│  |-head of state (P35)
...
│  |-elector (P2319)
│  |=elected in (P2715)
...
│  ├──Wikidata property for occupations (Q24043375)
│  │   -field of this occupation (P425)
...
│     ├──Wikidata property for items about people (Q18608871)
│     │  |=place of birth (P19)
│     │  |=place of death (P20)
│     │  |=position held (P39)
...
│     │  |=educated at (P69)
...
│     │  |-political party (P102)
...
│     │  |=occupation (P106)
│     │  |=employer (P108)
...
│     │  |=name (P2561)
...
│     └──Wikidata property for items about organizations (Q18608993)
...
│        |-founded by (P112)
...
│        |-headquarters location (P159)
...
│        |=dissolved, abolished or demolished (P576)
...
```

Figure 7. A portion of Wikidata property taxonomy, extracted by means of the Wikidata-taxonomy tool.

In order to individuate the domain and range of a given property, we considered the *type constraint* (Q21503250), representing the domain of the property, and the *value type constraint* (Q21510865), representing the range of the property. In some cases, we used the additional information about properties provided by the *Property Talk* tool (www.wikidata.org/wiki/Template:Property_talk, accessed on 10 May 2021), to help us understand the correct meaning of the property and the way it is used, considering the examples of usage and the discussion between users available in the *Property Talk* (it is possible to view the *Property Talk* associated to a particular property using the base URL *https://www.wikidata.org/wiki/Property_talk:PWID* (accessed on 10 May 2021) replacing "PWID" with the Wikidata Property ID of the property which we want to analyze).

We analyzed the list of extracted properties to select the candidates to be mapped to HERO. Some properties, such as *place of birth* (P19) and *place of death* (P20), can be directly mapped to the corresponding HERO properties (e.g., *hasBirthPlace* and *hasDeathPlace*, respectively). These cases are captured by mappings such as mapping 2 in Table 3. For instance, the Wikidata statement <*Sandro Pertini, place of birth (P19), Stella San Giovanni*> is translated in HERO as <*SandroPertini, hasBirthPlace, StellaSanGiovanni*>.

In some cases, we defined more complex mappings in order to represent in HERO the rich information provided by some description patterns in Wikidata. In particular, some of these complex mappings take into account the Wikidata *qualifiers*. In Wikidata, a *qualifier* is a property used to refine a property statement. For instance, Figure 8 shows the Wikidata statement asserting that Sergio Pininfarina (Q286469)—an Italian businessman and designer—has been employed (*employer* property, P108) at Politecnico di Torino; this information is refined by three qualifiers: *start time*, *end time*, and *position held*, to set time boundaries (1974–1977) and to specify his role (*professor*).



employer

Politecnico di Torino

start time 1974

end time 1977

position held professor

Figure 8. An example showing the use of qualifiers to represent additional information about a specific statement in Wikidata (from https://www.wikidata.org/wiki/Q286469, accessed on 10 May 2021).

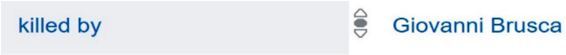
In such cases, the translation is performed according to mapping patterns such as mapping 3 in Table 3. Intuitively, in such mapping patterns, universally quantified variables (i.e., those occurring in the Wikidata pattern) represent corresponding individuals in Wikidata and in the PRiSMHA Semantic KB, while existentially quantified variables (i.e., those occurring only in the HERO pattern) represent HERO individuals that must be created (or retrieved from the PRiSMHA Semantic KB, if they are already present in it) when applying the mapping. It is worth noting that there are information items that are conceptually implied by Wikidata patterns but can not be formally derived from them: Complex mappings make them explicit in terms of HERO. For example, if someone is employed in an organization, playing a specific role in it, such an organization actually *institutionalizes* (in the sense specified in [10]) that role (see Section 2).

We describe the semantics of mapping 3 in Table 3 through the example of Sergio Pininfarina. Given the Wikidata description shown in Figure 8, the application of the mappings prescribes to enrich the PRiSMHA Semantic KB by:

- Stating that Sergio Pininfarina is affiliated to Politecnico di Torino (<*SergioPininfarina, isAffiliatedTo, Polito*>);
- Creating an instance of the *Profession* HERO class, *professorAtPolito*, to represent the specific role played, i.e., "professor at Politecnico di Torino" (<*professorAtPolito, rdf:type, Profession*>);
- Stating that Sergio Pininfarina played the role of professorAtPolito (<SergioPininfarina, isClassifiesdBy, professorAtPolito);
- Stating that the role *professorAtPolito* is institutionalized by the Politecnico di Torino (<*Polito, institutionalizes, professorAtPolito*>);
- Stating that the *professorAtPolito* role is a sub-role of the *professor* role (<*professorAtPolito, subRoleOf, professor*>); in fact, since the Wikidata *professor* (Q121594) category is mapped to the *professor* HERO role (i.e., *Map-c(Q121594) = professor*), in this case, *professor* is an instance of the *Role* HERO class; thus, the conditions expressed in mapping 3 (Table 3) are satisfied;
- Creating an instance of the *TemporaryAffiliationToOrganization* HERO class, *ta*, to represent the fact that Sergio Pininfarina was temporarily affiliated to the Politecnico di Torino (<*ta, rdf:type, TemporaryAffiliationToOrganization*>);
- Stating that Sergio Pininfarina was temporarily affiliated to some organization (<*ta, hasAffiliationToOrganizationEntityElement, SergioPininfarina*>);
- Stating that the organization Sergio Pininfarina was temporarily affiliated to was the Politecnico of Torino (<*ta, hasAffiliationToOrganizationOrganizationElement, Polito*>);
- Stating that the temporary affiliation lasted for a specific time interval *temp* (<*ta, hasTemporalParameterElement, temp*>);

- Creating an instance *tc* of the *TemporaryClassification* HERO class, to represent the fact that Sergio Pininfarina temporarily played the role of professor at Politecnico di Torino (<*tc, rdf:type,TemporaryClassification*>);
- Stating that Sergio Pininfarina has temporarily played some role (<*tc, hasClassificationClassifiedEntityElement, SergioPininfarina*>);
- Stating that the role temporarily played by Sergio Pininfarina is *professorAtPolito* (<*tc, hasClassificationClassifyingConceptElement, professorAtPolito*>);
- Stating that the temporary classification lasted for the *temp* time interval (<*tc, hasTemporalParameterElement, temp*>);
- Stating that *temp* is an instance of the *TimeInterval* HERO class (<*temp rdf:type TimeInterval*>);
- Stating that *temp* starts in 1974 (<*temp, intBeginsIn, 1974*>);
- Stating *temp* ends in 1977 (<*temp, intEndsIn, 1977*>).

In some cases, we defined complex mappings from Wikidata description patterns to event representations in HERO. For instance, Figure 9 shows a Wikidata statement, involving the *killed by* Wikidata property (P157), asserting that Giovanni Falcone (an Italian magistrate murdered by the Mafia) was assassinated by Giovanni Brusca, an Italian mafioso (among other killers).


killed by

Giovanni Brusca

Figure 9. An example showing a Wikidata statement about Giovanni Falcone (from https://www.wikidata.org/wiki/Q207073, accessed on 10 May 2021).

In such cases, the Wikidata descriptions are translated by explicitly introducing HERO events (see mapping 4 in Table 3).

In our example, the system suggests adding two events: (i) a homicide, perpetrated by Giovanni Brusca (among the others) with Giovanni Falcone as a victim, and (ii) the death of Giovanni Falcone, which was caused by the homicide. The choice of suggesting both events (the homicide and the death) is the result of our analysis of the intended semantics of the Wikidata property *killed by*: The use of such a property, in fact, implies that the subject died as a consequence of a homicide.

The application of the mappings prescribes to enrich the PRiSMHA Semantic KB by:

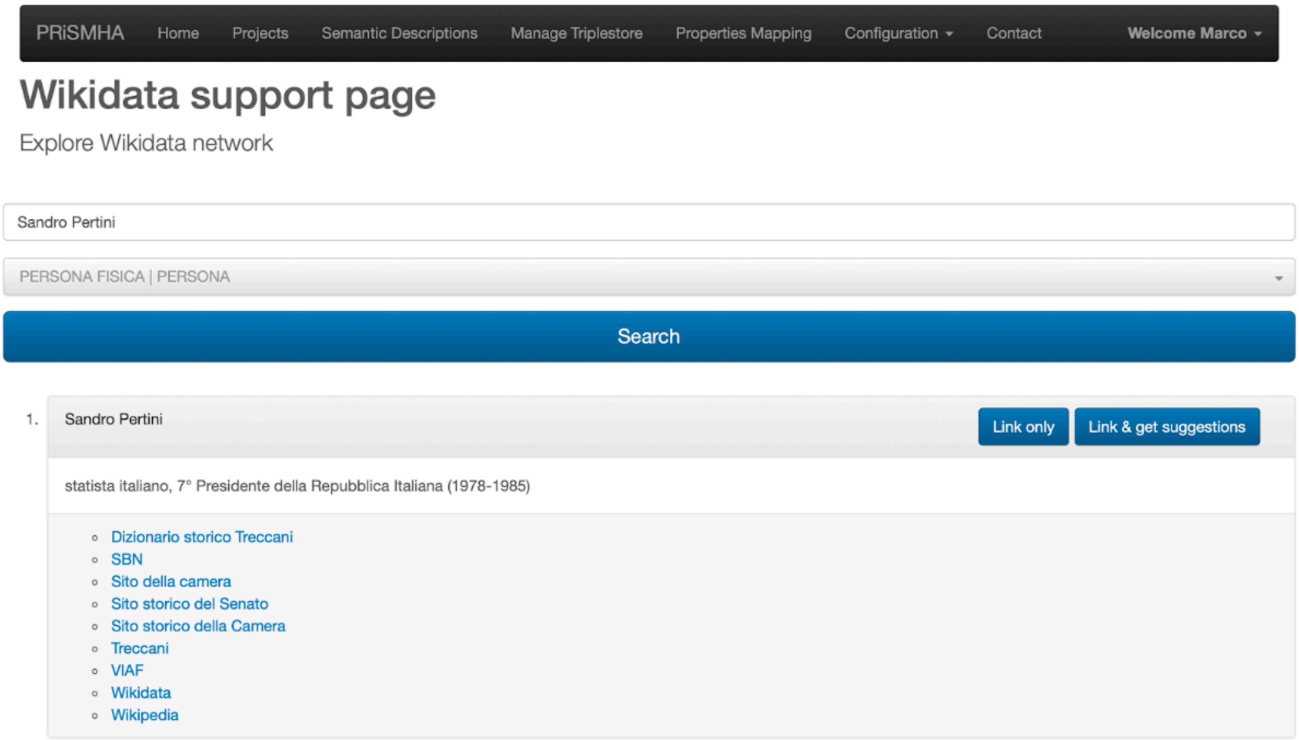
- Creating an instance *e1* of the *Homicide* class (<*e1, rdf:type, Homicide*>) (the use of the *Homicide* HERO class in the definition of this mapping is a choice based on the semantics of the Wikidata property *killed by*, emerged by our analysis);
- Stating that Giovanni Falcone "participated" in the homicide *e1* with the role of "patient", i.e., the participant who was affected by the event (<*e1, hasPatient, GiovanniFalcone*>);
- Stating that Giovanni Brusca participated in the homicide *e1* with the role of "agent", i.e., a participant who voluntarily acted in the event (<*e1, hasAgent, GiovanniBrusca*>);
- Creating an instance *e2* of the *PersonDeath* class (<*e2, rdf:type, PersonDeath*>.
- Stating that Giovanni Falcone "participated" in the event of his death (*e2*) again with the role of "patient" (<*e2, hasPatient, GiovanniFalcone*>);
- Stating that the homicide of Giovanni Falcone caused his death (<*e1, isCausalFactorOf, e2*>, where *isCausalFactorOf* is a HERO property representing a causality relation between events.

As far as mappings between properties and mappings between complex patterns are concerned, we defined a total of 57 simple property mappings and 32 complex mappings.

4. Using Wikidata Information to Suggest Semantic Representations

Imagine a user is annotating a document, and she needs to add a new entity, corresponding to Sandro Pertini, to the Semantic KB. As described in Section 2, she can click *View or add annotations* (Figure 2) to get the window enabling her to add a new entity (by clicking *Add (and link) new entity*: Figure 3). The user can now select the entity type, *PhysicalPerson* in our example, and provide a label, e.g., "Sandro Pertini": This makes the system generate the form to be filled in to characterize the new entity (Figure 4); the form containing properties suitable to describe the new entity is dynamically generated on the basis of the selected type (class) for the entity itself (see [4]). At this point, the user can ask for suggestions from external resources (Wikidata in the current prototype) by clicking *Search on external resource*.

The system opens a new window (Figure 10) with a small form already filled in with a type (e.g., *PhysicalPerson*) and a label to be used to search Wikidata. By clicking the *Search* button, the system searches Wikidata for entities whose type and label are the specified ones. As far as the type is concerned, the system searches for all Wikidata entities whose category corresponds to the selected HERO class (*PhysicalPerson* in our example). More precisely, it searches for all the triples in which the property is *instance of* (P31) and the object is a Wikidata category *c* corresponding to the selected HERO class, or a subcategory of *c* (see [6])].



Wikidata support page

Explore Wikidata network

Sandro Pertini

PERSONA FISICA | PERSONA

1. Sandro Pertini

statista italiano, 7° Presidente della Repubblica Italiana (1978-1985)

- Dizionario storico Treccani
- SBN
- Sito della camera
- Sito storico del Senato
- Sito storico della Camera
- Treccani
- VIAF
- Wikidata
- Wikipedia

Figure 10. The Wikidata support page. In this example, we looked for an entity of type *PhysicalPerson* and labeled "Sandro Pertini".

The results of the Wikidata search are shown below the form: Figure 10 shows the single candidate found in our example. Each candidate is displayed within a "card", showing the Wikidata label, a short Italian description, and the links to the entity representation in external resources.

In the upper part of each card, two buttons are available: The *Link only* button simply adds the link between the entity that we are creating (e.g., Sandro Pertini) and the selected Wikidata entity (Q1233), using the property *skos:exactMatch* (www.w3.org/TR/skos-reference/#L4858, accessed on 10 May 2021), corresponding to the *Corrisponde esattamente a* "property" in the form (see Figure 4); the *Link & get suggestions* button, besides linking the two entities, actually activates the generation of suggestions.

When the user clicks the *Link & get suggestions* button, the system extracts from Wikidata all the statements containing mapped properties and referring to the entity in focus, Sandro Pertini in the current example; part of this information is graphically shown in Figure 11.

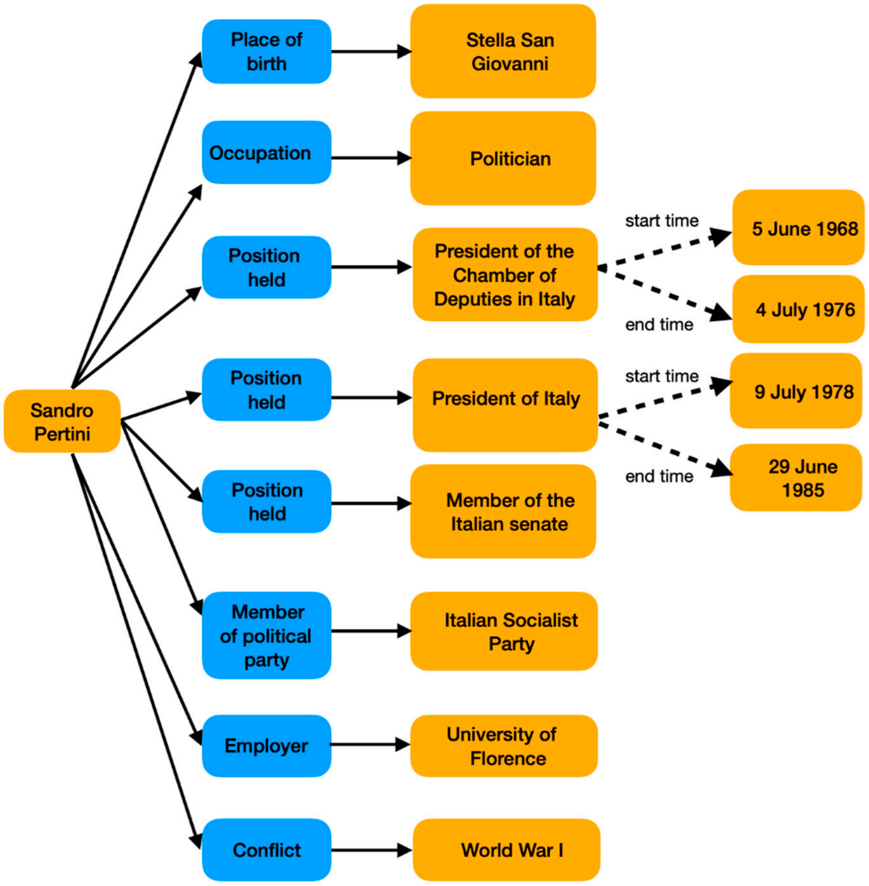


Figure 11. A portion of the network extracted by looking for information about Sandro Pertini in Wikidata. Orange nodes are entities while blue nodes are properties in Wikidata. Dashed arrows represent qualifiers.

In SPARQL terms, we first ask for all the properties that describe Sandro Pertini (Q1233) by means of the following query (the Wikidata ID at line 2 is the ID of the entity we are searching information for):

```
1 select distinct ?p where {
2     wd:Q1233 ?p ?o.
3 }
```

From this set of properties, we extract the ones for which a mapping is defined. Then, for each selected property, considering the Wikidata entity in focus as subject, we look for the available fillers (objects), by means of SPARQL queries such as the following:

```
1 select distinct ?o where {
2 wd:Q1233 p:P39 ?os.
3 ?os ps:P39 ?o.
4 }
```

At line 2, using the *p* namespace enables us to obtain the statements related to a specific Wikidata property (P39—*position held* in the example), having the entity Q1233 (*Sandro Pertini*) as subject. At line 3, using the *ps* namespace enables us to obtain the fillers (objects) for the selected property (e.g., *President of Italy*, *Member of the Italian Senate*, etc.).

When all this information has been extracted, the system applies the mappings (Section 3) and produces the suggestions for Sandro Pertini:

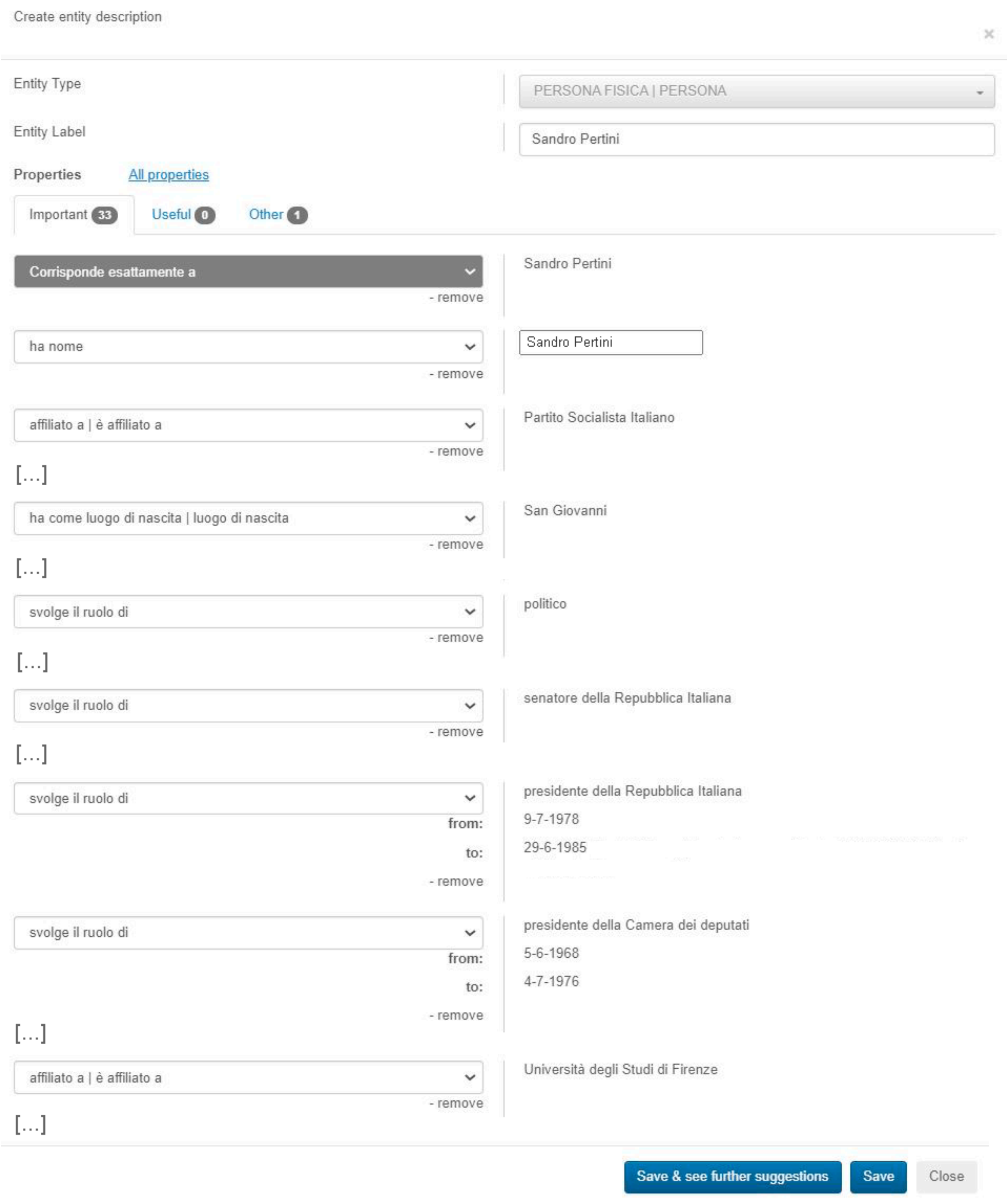
- Is a Person;
- Has name Alessandro Pertini, Sandro Pertini;
- Has family name Pertini;
- Was born on 25-9-1896;
- Was born in Stella San Giovanni;
- Died on 24-2-1990;
- Died in Rome;
- Was married to Carla Voltolina;
- Played the role of student at University of Genoa;
- Played the role of politician;
- Played the role of journalist;
- Played the role of member of the Constituent Assembly of Italy;
- Played the role of President of the Chamber of Deputies of Italy from 5-6-1968 to 4-7-1976;
- Played the role of President of Italy from 9-7-1978 to 29-6-1985;
- Played the role of member of the Chamber of Deputies of the Italian Republic;
- Played the role of member of the Italian Senate;
- Played the role of Italian senator for life from 29-6-1985 to 24-2-1990;
- Was affiliated to United Socialist Party;
- Was affiliated to Italian Socialist Party;
- Was affiliated to University of Genoa;
- Was affiliated to University of Florence;
- Participated in World War I;
- Was a candidate in the 1978 Italian presidential election.

The form in Figure 12 shows some of the suggestions.

Suggestions can be refused by clicking the *remove* button under each property. Entities suggested as property fillers (e.g., *Italian Socialist Party*—Q590750, suggested filler of an *affiliatedTo* HERO property) can be new entities or entities that are already present in the PRiSMHA Semantic KB. In the former case, the system suggests their creation on the basis of the Wikidata label and type. Moreover, the new PRiSMHA entity is linked to the Wikidata entity by means of the *skos:exactMatch* relation. In the latter case, the system checks if there is a statement in the PRiSMHA Semantic KB in which the property is *skos:exactMatch* and the object is the Wikidata entity (e.g., *Italian Socialist Party*): The subject of such a statement is the corresponding PRiSMHA entity to be suggested.

Some of the suggestions correspond to complex patterns, which are described in Section 3.2. For example, Sandro Pertini played the role of President of Italy from 9/7/1978 to 29/6/1985. It is worth noting that this information is shown in a fairly simple shape to users (Figure 12), although it corresponds to a complex representation in HERO terms, analogous to mapping 3 in Table 3. This is a way in which PRiSMHA hides the complexity of the ontology, without abdicating the full expressive power of the formal semantic representation underneath it (see also Section 6 for a discussion).

After having evaluated all suggestions (and possibly removed and replaced them with different information), the user can simply save the form by clicking the *Save* button, or she can save the form and look at further suggestions by clicking the *Save & see further suggestions* button. In the latter case, all information in the form is saved (i.e., the corresponding triples are added to the Semantic KB, as indicated by the mappings, and the user is prompted with a new window displaying further suggestions, if any (Figure 13)).



Create entity description

Entity Type: PERSONA FISICA | PERSONA

Entity Label: Sandro Pertini

Properties All properties

Important 33 Useful 0 Other 1

Corrisponde esattamente a — Sandro Pertini
- remove

ha nome — Sandro Pertini
- remove

affiliato a | è affiliato a — Partito Socialista Italiano
- remove

[…]

ha come luogo di nascita | luogo di nascita — San Giovanni
- remove

[…]

svolge il ruolo di — politico
- remove

[…]

svolge il ruolo di — senatore della Repubblica Italiana
- remove

[…]

svolge il ruolo di — presidente della Repubblica Italiana
from: 9-7-1978
to: 29-6-1985
- remove

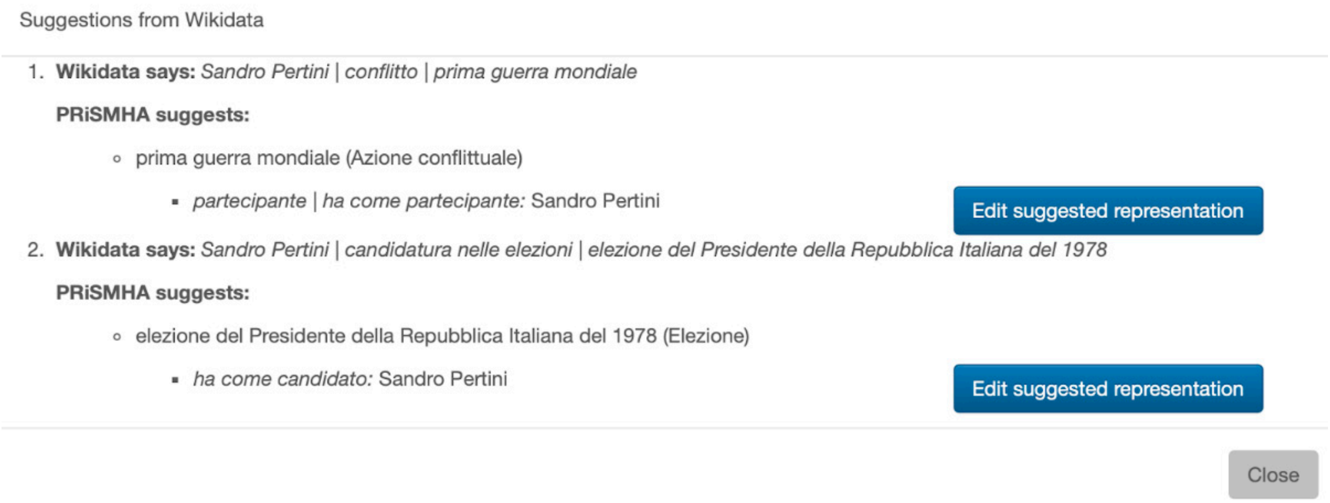
svolge il ruolo di — presidente della Camera dei deputati
from: 5-6-1968
to: 4-7-1976
- remove

[…]

affiliato a | è affiliato a — Università degli Studi di Firenze
- remove

[…]

Figure 12. The form to characterize Sandro Pertini, filled in with suggestions obtained from Wikidata.



Suggestions from Wikidata

1. **Wikidata says:** *Sandro Pertini | conflitto | prima guerra mondiale*

   PRiSMHA suggests:

   - prima guerra mondiale (Azione conflittuale)
     - *partecipante | ha come partecipante:* Sandro Pertini

2. **Wikidata says:** *Sandro Pertini | candidatura nelle elezioni | elezione del Presidente della Repubblica Italiana del 1978*

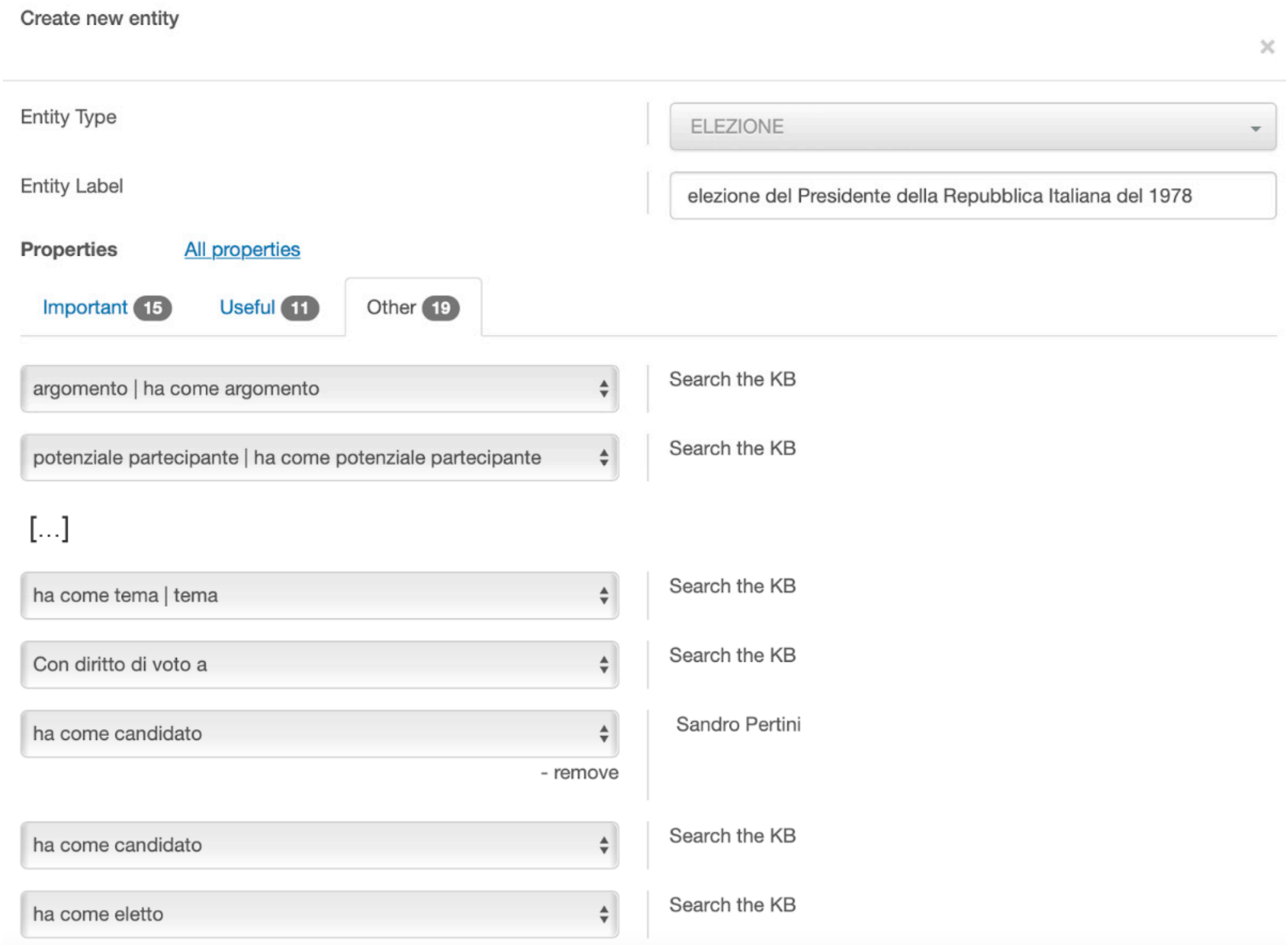
   PRiSMHA suggests:

   - elezione del Presidente della Repubblica Italiana del 1978 (Elezione)
     - *ha come candidato:* Sandro Pertini

Figure 13. Event suggestions for Sandro Pertini, obtained from Wikidata.

Figure 13 shows two further suggestions related to Sandro Pertini: his participation in World War I and his participation in the 1978 Italian presidential election as a candidate. Underneath this simple user interface, again, there is the representation of events, based on HERO, as described in Section 2. For example, the first suggested event (World War I) corresponds to the following triples: <*e*, *rdf:type*, *War*> <*e*, *hasParticipant*, *SandroPertini*> <*SandroPertini*, *rdf:type*, *PhysicalPerson*>.

The user can edit the representation of a suggested event by clicking the corresponding *Edit suggested representation* button. Consider the second suggested event, in our example: When clicking the *Edit suggested representation* button, the user is prompted with a new form for characterizing the event (Figure 14 shows a part of it), which is filled in with information from Wikidata: In the example, the type (i.e., the *Election* HERO class), the label ("1978 Italian presidential election"), and the *hasCandidate* HERO property, filled in with the entity *Sandro Pertini* (this specific suggestion is derived from the complex mapping of the *candidacy in election* Wikidata property, P3602, onto the complex HERO pattern representing specific participation in election events as a candidate).



Create new entity ×

Entity Type: ELEZIONE

Entity Label: elezione del Presidente della Repubblica Italiana del 1978

Properties All properties

Important 15 Useful 11 Other 19

argomento | ha come argomento	Search the KB
potenziale partecipante | ha come potenziale partecipante	Search the KB
[…]	
ha come tema | tema	Search the KB
Con diritto di voto a	Search the KB
ha come candidato - remove	Sandro Pertini
ha come candidato	Search the KB
ha come eletto	Search the KB

Figure 14. Part of the form related to the characterization of the 1978 Italian presidential election, with Sandro Pertini as candidate.

5. Evaluation and Discussion

The aim of the evaluation we discuss in this section is to assess our mappings, in particular the complex ones (see Section 3.1) with respect to the quality of the suggestions for the users of the annotation platform. Obviously, the quantity (but probably also the quality) of the information we can extract from Wikidata strongly depends on the entity we are searching for. For the evaluation, we need entities, within the selected domain, for which we can get a significant amount of information, and, in particular, information enabling the system to use the complex mappings we aim at assessing. Thus, we selected four famous preeminent figures of the Italian history of the 20th century for which the information we can extract from Wikidata satisfy our requirements: Sandro Pertini, Aldo Moro, Sergio Garavini, and Giovanni Falcone.

We prepared a questionnaire presenting the information extracted for each person and asking user opinions through questions including radio buttons and free-text comments (see Appendix A). Then, we submitted the questionnaire to a pool of potential users of the PRiSMHA platform, obtaining complete answers from 37 participants: 49% of them were between 40 and 59 years old, 24% were between 20 and 39 years old,, and 27% were between 60 and 79 years old,; 59.5% had a master degree, 32.5% had a PhD, and 8% a high school degree; 81% were workers, 11% were retired, and 8% were students.

As we claimed above, our goal was a qualitative evaluation of the suggestions, i.e., of the information extracted from Wikidata on the basis of our mappings; therefore, a quantitative analysis of the results makes little sense, since data are too small. However, we provide a short comment on quantitative results since they can still provide us with useful insights.

As shown in Table 4, overall, the information extracted from Wikidata on the basis of our mappings seems to be useful: More than 80% of it was accepted by users (and the most part was accepted without changes). However, a significant number of answers indicate that the extracted information is not enough to characterize the person in focus (*missing info* row in Table 4), with slightly different results for the four persons: 49% for Pertini, 62% for Moro, 35% for Garavini, and 65% for Falcone. For Moro, some users also claimed that there is too much information (*too much info* row in Table 4): To understand this twofold claim, we need to look at user comments, which are discussed below.

Table 4. Results (percentage of answers). Options refer to Appendix A.

	Pertini	Moro	Garavini	Falcone	Overall
used info (as it is or with modification)—*options [a]+[b]*	82%	81%	82%	83%	82%
missing info—*option [f]*	49%	62%	35%	65%	53%
too much info—*option [h]*	11%	41%	0%	0%	13%
too little info—*option [i]*	22%	19%	30%	62%	33%

The core of our evaluation is user comments. In the following, we first present, and then discuss, the most relevant issues that emerged from them.

All free-text questions (see Appendix A) were optional, and besides asking users for the reasons for their radio-button choices, they asked for explanations about missing or too much/too little information (see Table 4). Only the last question encouraged users to express a completely free comment. Some users explicitly wrote that the suggestions were clear and precise. In particular, a participant wrote: "It seems to me that the system proposes clear and accurate information, easy to understand, also without a specific knowledge of the persons in focus or of the historical context they belong to. In some cases, the information is incomplete, but fortunately the proposal can be changed and integrated. I did not find really wrong or useless suggestions".

However, the most interesting comments are those containing some criticism. We discuss them in the following.

(1) A lot of participants (54%) remarked that the information about Pertini lacks data about the very important fact that he was a Partisan in the Italian Resistenza, fighting against Fascism during World War II. Moreover, a significant number of users (24%) reported, more specifically, that he was a member of the CLN (*Comitato di Liberazione Nazionale*), he was kept in prison (from where he escaped), he was exiled in France, he was director of the *Avanti!* newspaper, all relevant data for the characterization of his life, according to users, and missing from our suggestions.

(2) Many participants (32%) wrote that the information suggested for Moro and referring to his different roles in different Italian Governments was too scattered and redundant, due to the specification, as distinct items, of all the periods of times in which he played each role (e.g., *Prime Minister of Italy*, *Italian Minister of Foreign Affairs*, and so on). Moreover, a significant number of users (54%) stated that suggestions about Moro miss information about his kidnapping and details about his homicide, in particular when and where it took place, and, most important, that he was killed by *Brigate Rosse* (the well-known Italian terrorist organization). Finally, some participants (11%) remark that his important role in the discussion about the *Compromesso Storico* (the possible political agreement between *Democrazia Cristiana* and *Partito Comunista* parties in the 1970s) is missing.

(3) Garavini was the less known figure we proposed, and Wikidata itself contains less data with regard to the other analyzed persons: For instance, a couple of users reported missing information about his education and his family. A significant

number of participants (19%) notice that he had an important role in CGIL Trade Union, and this information is missing. A single, but very important, comment underlines that the birth date of Garavini is wrong (Wikidata says 8 April 1927—www.wikidata.org/wiki/Q338536, accessed on 10 May 2021), while the correct date is 18 May 1926—it.wikipedia.org/wiki/Sergio_Garavini, accessed on 10 May 2021).

(4) A lot of comments (51% of users) report that suggestions about Falcone lack information about his fight against the Mafia, and in particular, his role in important trials against such a criminal organization, such as the well-known *Maxiprocesso*. Moreover, a number of users (14%) complain about the lack of important details about his homicide, such as where and when it took place, or the fact that the killers were affiliated to the Mafia, or that other people died in that massacre, such as his wife (Francesca Morvillo) and his bodyguards. Finally, a few participants (8%) remark that his relationship with Paolo Borsellino (a well-known Italian magistrate who closely collaborated with Falcone) is totally missing.

(5) There are some general comments that do not refer to any specific person analyzed but represent interesting feedback for us. Some users remarked that the information about the affiliation of a person to a university, together with the information that he was a student, or a professor, at that university, is redundant. A user wrote that too many suggestions (such as, for instance, in the case of political roles for Moro) could take a long time to be checked, and this can become an overload instead of help for users.

A lot of remarks by participants refer to missing information. In many cases, this is not due to a limitation of our mappings but to the fact that the information is simply missing in Wikidata. This is the case for all the remarks about Pertini (point (1)); details about the homicide of Moro (and in particular, for the *Brigate Rosse* as killers), and his role in the discussion about the *Compromesso Storico* (point (2)); data concerning Garavini's education and family, as well as his role in CGIL Trade Union (point (3)); information about the fight of Falcone against the Mafia and his role in the *Maxiprocesso*, the fact that his wife and his bodyguards died when he was killed, and his relationship with Paolo Borsellino (point (4)). Finally, a comment underlined a mistake in Wikidata, i.e., Garavini's birth date.

The missing information that the killers of Falcone were affiliated to the Mafia deserves a slightly different consideration. The Mafia as an entity (a criminal organization—Q1458155) is not mentioned in the data directly related to Falcone (Q207073); the word "Mafia" only occurs in the textual description of the entity ("Italian magistrate murdered by the Mafia"). The same is true for three out of five mentioned killers (namely, La Barbera, Cancemi, and Ganci); for Brusca, the word is not used, while in the case of Riina, Wikidata says that his *family* (property P53) is Sicilian Mafia (instance of the category *criminal organization*). Currently, we do not access these data for the following reasons:

(a) We decided to consider only Wikidata triples where the entity in focus (Falcone in this case) is the subject, without extracting data about property fillers (i.e., objects).

(b) We have no mapping for the Wikidata *family* property (P53), i.e., no corresponding concept is present in HERO. Moreover, it does not seem to be suited to express affiliation to a criminal organization.

(c) At the moment, we do not exploit text in the descriptions of Wikidata entities we search for.

In particular, points (a) and (c) could be taken into consideration in our future work, in order to enhance suggestions.

The missing suggestion about Moro's kidnapping is also worth briefly discussing. The Wikidata data about Moro contain a reference to his kidnapping: *kidnapping of Aldo Moro* (Q1634609), instance of *kidnapping* (Q318296) is the filler of the *cause of death* property (P509). We do not grasp this information, since we decided not to map this property, due to the confusion related to its range, and in particular to the fact that the notion of cause underneath it is definitely not precise: For example, considering the kidnapping of Moro, the cause of his death is simply wrong. However, since this property can provide us with useful information, we will take it into consideration again for a deeper study.

Finally, some comments point out possible wrong choices in the presentation of the information to users. The information suggested for Moro referring to his different roles in different Italian Governments, which were considered scattered and redundant by many participants, could be presented in a more compact way. A general strategy could be to merge all data coming from the same property (e.g., *position held*) with the same filler (e.g., *Prime Minister of Italy*) but different qualifiers values (e.g., *start time*: … *end time*: …), presenting such data as a single suggestion.

Another case that recommends a different presentation concerns the information about affiliation to a university, when being a student (or a professor) at the same university. Our mapping for the *educated at* Wikidata property (P69) prescribes the assertion of both the fact that the subject played the role of student at a given University and that the same subject was affiliated to that University (see Section 3.1). The second assertion is more generic than the first one; thus, users tend to consider it useless, but we think that it can be useful for the system. The solution could be hiding the affiliation relation when presenting suggestions to the user, while keeping it in the system KB (if the user confirms the correctness of the overall suggestion).

We conclude this section by briefly commenting on a participant observation that too many suggestions could turn into an overload for users. Trying to provide more compact suggestions could slightly mitigate this problem. However, obviously, the trade-off between helping and bothering users with supporting features is a delicate balance to be found: Our approach, based on a careful in-depth analysis of Wikidata aiming at extracting only high-quality data, "filtered" by HERO, aims at finding such a balance.

6. Related Work

Ontology-based annotation and ontology-based data access. Ontologies are a formal representation of categories, properties, and relations between concepts [15]. An ontology is usually employed to extensively represent a specific domain, thus providing for it a semantic vocabulary, but it also defines a schema that can be interconnected with already developed cross-domain knowledge bases, in order to achieve interoperability [16]. One of the multiple applications of an ontology is to serve as a semantic model for document annotation in those scenarios where users are required to enrich a document by exploiting the vocabulary provided by the ontology. For example, in the annotation framework proposed by Andrews and colleagues [17], ontologies are considered as the reference vocabulary, and the annotation process links the document to the semantic model by associating entity mentions to categories in the ontology. The proposed approach assumes that the ontology exhaustively represents the domain, including individuals, and annotators simply have to provide a link from the document to an element already available in the ontology. Consider, for example, a document about the magistrate Giovanni Falcone, telling that Giovanni Falcone was assassinated by Giovanni Brusca: The aforementioned framework assumes that the ontology already contains the characterization of Giovanni Falcone and Giovanni Brusca (and maybe also of the assassination), and the user is called to link the document with such entities.

In PRiSMHA, we do not assume that all individuals that could be mentioned in archival documents are already present in the knowledge base, and we ask users of the platform to add and characterize new entities when needed for the annotation. Since this activity is time-consuming, the hypothesis we are presenting in this paper is to exploit external resources, such as Wikidata, to get information suitable for the characterizations of the new entities, such as Giovanni Falcone and Giovanni Brusca.

A lot of works investigated the possibility of exploiting the semantic power of ontologies to positively impact on the semantic (meta)data accessibility [5,18–20]. In PRiSMHA, ontology-based (meta)data access turns into ontology-based (meta)data production. In this respect, one of the goals of the project is to provide users with a platform that effectively supports them in the annotation process, hiding the complexity of the ontology, while, at the same time, offering them the full expressive power of the semantic schema. This paper provides a contribution in this direction: In order to alleviate the burden of ontology-based annotation (and, in particular, of the creation and characterization of new entities), PRiSMHA provides the users of the annotation platform with automatically extracted suggestions that can be simply sifted through, thus not only saving time but also simplifying the task (see Section 5).

Using Wikidata. The idea of exploiting Linked Open Data [21] to get historical and cultural data is not new (see, for instance, [22]). A hub in the LOD cloud (lod-cloud.net) is Wikidata [7], born in 2012 as a central storage repository for the Wikimedia project, it represents structured data and makes them freely accessible for visualization, extraction, and modification in human- and machine-readable formats. In Wikidata, the entries represent concepts or objects. The knowledge is stored as triples, which describe entries and employ properties to characterize them. Since the beginning, Wikidata has been perceived as a linking hub for domain-specific knowledge bases [23,24]; in particular, Wikidata is an important point of convergence for the Cultural Heritage world in canonical [25,26] and contemporary fashion, such as the *Social Networks and Archival Context* project (snaccooperative.org), which aims at making the relations among persons, families, and organizations explicit, also by exploiting Wikidata. Additionally, to emphasize the centrality of Wikidata for the Cultural Heritage world, Europeana published the list of practical steps data providers can take to upload and align their vocabularies with Wikidata (pro.europeana.eu/post/why-data-partners-should-link-their-vocabulary-to-wikidata-a-new-case-study, accessed on 10 May 2021).

Wikidata has been widely employed, along with ontologies in two directions: (i) Wikidata as an enhancement for local metadata [27–29]; (ii) Wikidata as a knowledge hub, where domain knowledge can be injected so to enrich the general knowledge base [26,29–31].

The direction closest to the approach described in this paper is the enhancement of local metadata on the basis of Wikidata. Recently, several efforts have been devoted to enrich local knowledge bases exploiting Wikidata. For example, van Veen and colleagues improved the access to a collection of Dutch historical newspapers by linking named entities mentioned in the news to corresponding Wikidata entries [32]. Once the first step of named entities recognition has been performed, the authors exploited machine learning to directly match entity mentions to Wikidata elements. Cooney relies on Wikidata to enhance the list of Holocaust-era ghettos [27]. The project is aimed at enriching the European Holocaust Research Infrastructure ghettos' authority by matching entries in the project dataset and categories in Wikidata through geographical information and names of ghettos. Opasjumruskit and colleagues tackle the problem of automatically augmenting ontologies in the framework of information extraction processes [28]. The authors start with a domain specific ontology, defined by domain experts, and enrich it on the basis of external knowledge bases, such as WordNet [33] and Wikidata. In order to enrich the ontology, the authors search Wikidata by using ontology class names and then match the results with concepts previously identified thanks to WordNet. More recently, van Veen and colleagues have emphasized the transition from a world made of multiple local identifiers to a world in which the unique knowledge base identifier is provided by Wikidata [34].

Unfortunately, none of the aforementioned works aims at building complex mappings between the involved semantic models as the ones addressed in this paper (Section 3), in which complex patterns are involved.

Ontology matching. Ontology matching [13] is the task of discovering and representing semantic correspondences among entities of different ontologies (usually two, but sometimes more). Such correspondences can be undirected or directed. In the latter case, they are usually called *mappings*. The need of ontology matching originates from the presence of different and overlapping ontologies in contexts where distinct pieces of information must be shared, integrated, translated in a common language or put in correspondence with each other. In many cases, this is a complex and expensive task; thus, its automatization is the subject of an active research field [35].

A semantic correspondence between ontology entities can be simple or complex [36]. In the former case, the correspondence expresses a relation between two simple entities, such as named classes, properties, or individuals; for instance, it may state that a named class in an ontology is equivalent to a named class in another ontology. In the latter case, the semantic correspondence refers to, at least, one complex expression, i.e., an expression composed of constructors or transformation functions applied to simple entities and/or to other complex expressions; it may state, for instance, that the set of individuals having some values for a property in an ontology is equivalent to the (set denoted by the) union of two classes in another ontology. The correspondences usually express equivalence, subsumption, disjointness, or some general relation.

In the case of PRiSMHA, the reason for matching the HERO ontology with the Wikidata semantic model is to automatically recognize in the latter those pieces of information that also belong to the domain covered by HERO and to provide users with their translation in the HERO vocabulary. Thus, we are motivated by data translation needs [13], and our correspondences actually are *mappings* from the Wikidata ontology (the *source*) to HERO (the *target*).

We have four kinds of mappings:

(i) Mapping of Wikidata categories to HERO classes (Section 3.1). For instance, the Wikidata category *strike* (Q49776) is mapped to the HERO *Strike* class.

(ii) Mappings of Wikidata categories to HERO individuals (Section 3.1). For instance, the Wikidata category *student* (Q48282) is mapped to the HERO *student* role individual (note that, in our framework, we distinguish between the ontology and the data: The former provides the vocabulary by means of which the latter are expressed; an ontology can contain individuals, and here, we refer only to them, not to those described by data; this distinction, although sharp in HERO, is a little bit fuzzier in Wikidata, but since our methodology for finding such a kind of mappings always starts from HERO, this is not a problem).

(iii) Mappings of Wikidata properties to HERO properties (Section 3.1). For example, the Wikidata property *family name* (P734) is mapped to the HERO *hasLastName* property.

(iv) Mappings of Wikidata individual characterization patterns to HERO individual characterization patterns; examples are mappings 3 and 4 in Table 3.

Mappings of types (i), (ii), and (iii) are simple, while type (iv) mappings are complex (actually, mappings of type (i) are formally complex, since the Wikidata model represents the ground entities categories as individuals, while HERO represents them as classes; in Wikidata, the relation between a ground entity—e.g., Rome, Barack Obama, World War II—and its categories are expressed by the Wikidata *instance of* (P31) property, while in HERO, they are captured by the *rdf:type* property; therefore, the mapping of type (i) stating that Q49776 Wikidata category is equivalent to the HERO *Strike* class would take the form: $\exists P31.\{Q49776\} \equiv hero{:}Strike$, which is an instantiation of the *Class by Attribute Value* pattern [37]; however, in principle, this complexity could be circumvented by preliminary re-phrasing the Wikidata taxonomy in terms of classes.

As stated above (Section 3.1), we manually specified type (iv) mappings. Unfortunately, there was no chance to use any available automatic matching system. Some state-of-the-art matching systems rely on [38,39] or benefit from [40], the presence of shared instance data in the matched ontologies. In our case, we exploit the mappings to suggest users some possibly useful information that they can accept, revise, or refuse when they populate the system with instance data. Thus, the mappings should be available from the very beginning, when we cannot assume any shared instance data between our system and Wikidata. Other approaches are based on rule-based mechanisms that instantiate a set of pre-defined complex correspondence patterns [37]. However, the well-known patterns used by those (and other) systems, despite covering many important situations, are not complete. In particular, they do not satisfy the needs of our data translation task (for instance, it is easy to see that no pattern listed in [37] or in [38] covers the case specified in mapping 3 in Table 3).

In general, complex matching research has obtained important results in the last years. However, the performance of automatic matching systems on complex matching tasks are rather poor compared with their results on simple matching tasks, and it is known that automatic complex matching still remains extremely challenging [41].

Furthermore, it is worth noting that HERO offers some basic ontological notions that are needed to account for the considered domain, and are useful for reasoning about it; however, these are not usually present in cross-domain ontologies and are left as implicit in many datasets, including Wikidata. For example, this is the case of social roles and their relations [9], of the notions of role institutionalization by organizations, and of the affiliation of agents to organizations [10]. Many complex mappings that we defined do rely on these notions, and thus, they actually include an inference step that makes explicit some knowledge that is not even mentioned in Wikidata (either by the data or by the underlying model). This would be a further obstacle to the exploitation of any available ontology matching system as well as an interesting but hard challenge to automatic complex matching research.

Type (iii) mappings were manually specified (Section 3.1). Our automatic matching system (Section 3.1) performed poorly on properties. It is known that in some cases, ontology matchers have some problems with properties [41]. In our case, we conjecture that this phenomenon could be caused by some peculiarities of the HERO model, where some properties with specific meaning have labels containing common names, which are possibly misleading for our matching system. For instance, the HERO property *hasEndingEvent* (labeled as "ending event", "has as the ending event", "evento finale", "ha come evento finale"), whose intended meaning is a relation between perdurants (events or states) and events representing their conclusion (e.g., the end of World War II), has been wrongly stated as equivalent to the Wikidata property *final event* (P3967), which actually represents the final event of a competition. However, an in-depth analysis of this phenomenon is out of the scope of the present work, and we leave it as a future work.

For what concerns types (i) and (ii) mappings, we firstly automatically computed a set of candidate mappings, which we manually evaluated and completed. The automatic step was based on a language-based element-level matching system [13] that makes use of external resources (Section 3.1). The decision of implementing a simple matcher tailored to our needs instead of using any available matching systems is motivated by several reasons. Firstly, current state-of-the art matching systems still have problems in matching a domain ontology with a cross-domain one [41], which is exactly our case. Secondly, the different approaches by which HERO and Wikidata were built entail some significant ontological differences between the two semantic models. HERO is based on an accurate ontological analysis, which exploits both existing foundational and core ontologies and the domain experts' knowledge, which was aimed at providing a well-founded and sound framework for domain-specific instance data [4,42]. Differently, Wikidata [7] is a large-scale crowd-based project aimed at collecting and providing users with a free huge amount of data in any domain. In Wikidata, the community can contribute both to instance data and to the schema (i.e., to the underlying ontology). Conflicting data are explicitly admitted, and some kind of vagueness or any idiosyncratic position can be reasonably expected also at the ontological level. For instance, let us consider the Wikidata *part of* (P361) transitive property. Wikidata states that Barack Obama was *part of* the 109th United States Congress (www.wikidata.org/wiki/Q76, accessed on 10 May 2021). If we add the statement that Barack Obama's brain is part of Barack Obama (which would be a valid usage of the P361 property), we could derive that the brain of Barack Obama was part of the 109th United States Congress. This paradoxical conclusion results from the lack of distinction in Wikidata between the different notions of part–whole relations [43]. On the opposite, HERO distinguishes among a notion of parthood imported from formal mereology (that can express the relationship between Obama and his brain), membership (that can express the relationship between Obama and the 109th United States Congress, as far as the latter is considered as a collection) and affiliation (that can express the relationship between Obama and the 109th United States Congress, as far as the latter is considered as an organization). For what concerns the category of the 109th United States Congress, Wikidata states that it is a *legislative term* (Q15238777), which is a subcategory of *time interval* (Q186081). Thus, we paradoxically have that Obama is part of a time interval. HERO does not allow us to derive such a conclusion, since in HERO, the 109th United States Congress can be considered either a *Collective* or an *Organization*, which are both disjoint from the *TimeInterval* class.

Such ontological differences between HERO and the Wikidata ontology would make it difficult to exploit structure-level (either syntactic or semantic) [13] matching systems in our scenario.

7. Conclusions and Future Work

In this paper, we tried to answer the initial research question, i.e., to verify if information extracted from available external resources (Wikidata in the presented case study) provides users with useful suggestions in the creation of new entities used in an ontology-driven annotation process. We described the alignment of the underlying semantic models and the process used to extract information and generate suggestions.

The evaluation of the approach provided us with positive feedback. In particular, users considered the suggestions, based on the information extracted from Wikidata, useful, at least as a good starting point to characterize entities (e.g., persons) to be used in the ontology-based annotation of documents.

Users' comments contain interesting recommendations for the enhancement of the system that encourage the following: (a) broadening the "boundaries" of the information extraction from Wikidata by analyzing the feasibility of an approach that, besides taking into account the properties directly related to the entity at hand, also gathers data about the fillers of those properties (e.g., Falcone's killers); (b) exploiting the information available in Wikidata as textual descriptions (e.g., the description of Giovanni Falcone as an "Italian magistrate murdered by the Mafia"—www.wikidata.org/wiki/Q207073, accessed on 10 May 2021).

Several comments recommend us to redesign the way in which the system presents suggestions to its users. The first case, clearly represented by the roles played by Aldo Moro in different Italian Governments, drives us to design a more compact presentation of the available information, merging all data referring to the same property with the same filler (e.g., being the Italian Prime Minister), but different qualifiers values (e.g., identifying different periods of time). A second interesting case refers to the affiliation to an organization while playing a specific role in it (e.g., being a student at a university). Affiliation is a more generic assertion than role playing, and users judged it redundant and unnecessary. Thus, affiliation could be left behind the scenes by offering to the user a more compact suggestion, involving only the played role. This is a general issue, which amounts to devising criteria for partitioning the information gathered from external resources into a set of suggestions presented to the users and a set of information items that are "trivially entailed" by the former and that should be hidden. This is a complex issue that deserves further research, since the notion of "trivial entailment" is strongly related to the users' expertise and it is unlikely that it could be reduced to (trivial) logical inferences.

Another aspect that could be taken into account is represented by the mapping choices, which are driven by the goal of the system. For example, if the semantic descriptions were used to build a narrative structure that uses events as a backbone [44], when facing Wikidata property such as *spouse* (P26), instead of (or in addition to) the HERO property *marriedTo*, the system could suggest an event of type *Marriage* that took place in a given place at a given time with (at least) two participants both playing the role of *spouse*.

Finally, we are planning to improve the system by providing it with the capability of gathering information from multiple sources. As a first step in this direction, we will consider Wikidata links to other datasets in the LOD cloud in order to both integrate and refine the information provided by Wikidata.

Author Contributions: Conceptualization, D.C., A.G., M.L., D.M.; Methodology, D.C., A.G., M.L., D.M.; Software, D.C., M.L.; Validation, D.C., A.G., M.L., D.M.; Formal analysis, D.C., A.G., M.L., D.M.; Investigation, D.C., A.G., M.L., D.M.; Resources, D.C., A.G., M.L., D.M.; Data curation, D.C., A.G., M.L., D.M.; Writing—original draft preparation, D.C., A.G., M.L., D.M.; Writing—review and editing, D.C., A.G., M.L., D.M.; Visualization, D.C., A.G., M.L., D.M.; Supervision, A.G., D.M.; Project administration, A.G.; Funding acquisition, A.G.. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Fondazione Compagnia di San Paolo and Università di Torino within the PRiSMHA project (CSTO168023).

Data Availability Statement: Not applicable.

Acknowledgments: We would like to thank all PRiSMHA collaborators, and in particular Daniele Paolo Radicioni, for their valuable support in the project.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

Questions used for the evaluation.
For each suggestion:
[a] I would accept it as it is
[b] I would modify it (because it is wrong)
[c] I would modify it (because it is incomplete)
[d] I would refuse it (because it is wrong)
[e] I would refuse it (because it is useless)
For each person (X):
If you want, you can explain us your choices (referring to X)
  [free text]
Do you think that important information about X is missing? (If you want, you can write which information items are missing in the comment below)
  [f] yes, important information is missing
  [g] no, the most important items are there
If you want, you can tell us which information items about X are missing
  [free text]
Do you think that suggested information about X are...
  [h] too much
  [i] too little
  [l] the proper amount
If you want, you can tell us something more about the amount of information about X
  [free text]
At the end:
If you have something more to tell us, here you can do it freely!
  [free text]

References

1. Hogenboom, F.; Frasincar, F.; Kaymak, U.; De Jong, F. An Overview of Event Extraction from Text. In Proceedings of the Workshop on Detection, Representation, and Exploitation of Events in the Semantic Web, Bonn, Germany, 23 October 2011; pp. 48–57.
2. Goy, A.; Damiano, R.; Loreto, F.; Magro, D.; Musso, S.; Radicioni, D.P.; Accornero, C.; Colla, D.; Lieto, A.; Mensa, E.; et al. PRiSMHA (Providing Rich Semantic Metadata for Historical Archives). In Proceedings of the Contextual Representation of Objects and Events in Language, Bolzano, Italy, 21–23 September 2017.
3. Motta, E.; Buckingham Shum, S.; Domingue, J. Ontology-driven document enrichment: Principles, tools and applications. *Int. J. Hum. Comput. Stud.* **2000**, *52*, 1071–1109. [CrossRef]

4. Goy, A.; Colla, D.; Magro, D.; Accornero, C.; Loreto, F.; Radicioni, D.P. Building Semantic Metadata for Historical Archives through an Ontology-driven User Interface. *J. Comput. Cult. Herit.* **2020**, *13*, 1–36. [CrossRef]
5. Sevilla, J.; Casanova-Salas, P.; Casas-Yrurzum, S.; Portalés, C. Multi-Purpose Ontology-Based Visualization of Spatio-Temporal Data: A Case Study on Silk Heritage. *Appl. Sci.* **2021**, *11*, 1636. [CrossRef]
6. Colla, D.; Goy, A.; Leontino, M.; Magro, D.; Picardi, C. Bringing Semantics into Historical Archives with Computer-aided Rich Metadata Generation. *J. Comput. Cult. Herit.* **2021**. under review.
7. Vrandečić, D.; Krötzsch, M. Wikidata: A free collaborative knowledgebase. *Commun. ACM* **2014**, *57*, 78–85. [CrossRef]
8. Borgo, S.; Masolo, C. Foundational choices in dolce. In *Handbook on Ontologies*, 2nd ed.; Staab, S., Studer, R., Eds.; Springer: Berlin/Heidelberg, Germany, 2009; pp. 361–381.
9. Masolo, C.; Vieu, L.; Bottazzi, E.; Catenacci, C.; Ferrario, R.; Gangemi, A.; Guarino, N. Social Roles and their Descriptions. In Proceedings of the Knowledge Representation Conference, Palo Alto, CA, USA, 2–5 June 2004; Dubois, D., Welty, C., Williams, M.A., Eds.; AAAI Press: Cambridge, MA, USA, 2004; pp. 267–277.
10. Bottazzi, E.; Ferrario, R. Preliminaries to a DOLCE Ontology of Organisations. *Int. J. Bus. Process Integr. Manag.* **2009**, *4*, 225–238. [CrossRef]
11. Parsons, T. *Events in the Semantics of English: A Study in Subatomic Semantics*; MIT Press: Cambridge, MA, USA, 1990.
12. Goy, A.; Magro, D.; Rovera, M. On the Role of Thematic Roles in a Historical Event Ontology. *Appl. Ontol.* **2018**, *13*, 19–39. [CrossRef]
13. Euzenat, J.; Shvaiko, P. *Ontology Matching*, 2nd ed.; Springer: Berlin/Heidelberg, Germany, 2013.
14. Navigli, R.; Ponzetto, S. BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artif. Intell.* **2012**, *193*, 217–250. [CrossRef]
15. Gruber, T.R. A translation approach to portable ontology specifications. *Knowl. Acquis.* **1993**, *5*, 199–220. [CrossRef]
16. Alma'aitah, W.Z.; Talib, A.Z.; Osman, M.A. Opportunities and challenges in enhancing access to metadata of Cultural Heritage collections: A survey. *Artif. Intell. Rev.* **2020**, *53*, 3621–3646. [CrossRef]
17. Andrews, P.; Zaihrayeu, I.; Pane, J. A Classification of Semantic Annotation Systems. *Semant. Web* **2012**, *3*, 223–248. [CrossRef]
18. Kollia, I.; Tzouvaras, V.; Drosopoulos, N.; Stamou, G. A systemic approach for effective semantic access to cultural content. *Semant. Web* **2012**, *3*, 65–83. [CrossRef]
19. Tonkin, E.L.; Tourte, G.J.L. Using the crowd to update Cultural Heritage catalogue. In Proceedings of the Involving the crowd in future museum experience design, San Jose, CA, USA, 7–12 May 2016; pp. 1–6.
20. Windhager, F.; Mayr, E.; Schreder, G.; Smuc, M.; Federico, P.; Miksch, S. Reframing Cultural Heritage collections in a visualization framework of space-time cubes. In Proceedings of the Histo-informatics workshop, CEUR, Krakow, Poland, 12–16 July 2016; Volume 1632.
21. Heath, T.; Bizer, C. *Linked Data: Evolving the Web into a Global Data Space*; Morgan & Claypool: San Rafael, CA, USA, 2011.
22. Daif, A.; Dahroug, A.T.; López-Nores, M.; González-Soutelo, S.; Bassani, M.; Antoniou, A.; Gil-Solla, A.; Ramos-Cabrer, R.; Pazos-Arias, J.J. A Mobile App to Learn About Cultural and Historical Associations in a Closed Loop with Humanities Experts. *Appl. Sci.* **2019**, *9*, 9. [CrossRef]
23. Neubert, J. Wikidata as a linking hub for knowledge organization systems? Integrating an authority mapping into Wikidata and learning lessons for KOS mappings. In Proceedings of the European Networked Knowledge Organization Systems Workshop, CEUR, Thessaloniki, Greece, 21 September 2017; Volume 1937.
24. Bouscarrat, L.; Bonnefoy, A.; Capponi, C.; Ramisch, C. Multilingual enrichment of disease biomedical ontologies. In Proceedings of the Workshop on Multilingual Biomedical Text Processing, Marseille, France, 16 May 2020; pp. 21–28.
25. Allison-Cassin, S.; Scott, D. Wikidata: A platform for your library's linked open data. *Code4Lib J.* **2018**, 40.
26. Faraj, G.; Micsik, A. Enriching Wikidata with cultural heritage data from the COURAGE project. In Proceedings of the Research Conference on Metadata and Semantics Research, Rome, Italy, 28–31 October 2019; pp. 407–418.
27. Cooey, N. Leveraging Wikidata to Enhance Authority Records in the EHRI Portal. *J. Libr. Metadata* **2019**, *19*, 83–98. [CrossRef]
28. Opasjumruskit, K.; Peters, D.; Schindler, S. ConTrOn: Continuously trained ontology based on technical data sheets and Wikidata. *arXiv* **2019**, preprint.
29. Heberlein, R. On the Flipside: Wikidata for Cultural Heritage Metadata through the Example of Numismatic Description. In Proceedings of the IFLA WLIC Conference, Athens, Greece, 24–30 August 2019.
30. Lemus-Rojas, M.; Odell, J. Creating Structured Linked Data to Generate Scholarly Profiles: A Pilot Project Using Wikidata and Scholia. *J. Librariansh. Sch. Commun.* **2018**, *6*, 1–23. [CrossRef]
31. Radio, E.; Fletcher, K.; Athea, M. Creating and Using a Glacier Authority Index to Document Climate Change. *Cat. Classif. Q.* **2020**, *58*, 486–504. [CrossRef]
32. van Veen, T.; Lonij, J.; Faber, W. Linking named entities in Dutch historical newspapers. In Proceedings of the Research Conference on Metadata and Semantics Research, Göttingen, Germany, 22–25 November 2016; pp. 205–210.
33. Miller, G.A. WordNet: A lexical database for English. *Commun. ACM* **1995**, *38*, 39–41. [CrossRef]
34. van Veen, T. Wikidata. *Inf. Technol. Libr.* **2019**, *38*, 72–81. [CrossRef]
35. Otero-Cerdeira, L.; Rodríguez-Martínez, F.J.; Gómez-Rodríguez, A. Ontology Matching: A Literature Review. *Expert Syst. Appl.* **2015**, *42*, 949–971. [CrossRef]

36. Thiéblin, E.; Haemmerlé, O.; Hernandez, N.; Trojahn, C. Survey on Complex Ontology Matching. *Semant. Web J.* **2020**, *11*, 689–727. [CrossRef]
37. Ritze, D.; Meilicke, C.; Šváb-Zamazal, O.; Stuckenschmidt, H.A. Pattern-Based Ontology Matching Approach for Detecting Complex Correspondences. In Proceedings of the Workshop on Ontology Matching, CEUR, Washington, DC, USA, 25 October 2009; Volume 551, pp. 25–36.
38. Zhou, L.; Cheatham, M.; Hitzler, P. Towards Association Rule-Based Complex Ontology Alignment. In Proceedings of the Joint International Semantic Technology Conference, LNCS, Hangzhou, China, 25–27 November 2019; Wang, X., Lisi, F., Xiao, G., Botoeva, E., Eds.; Springer: Berlin/heidelberg, Germany, 2020; Volume 12032, pp. 287–303.
39. Zhou, L.; Hitzler, P. AROA Results for OAEI 2020. In Proceedings of the Workshop on Ontology Matching, CEUR, Athens, Greece, 2–6 November 2020; Volume 2788, pp. 161–167.
40. Lima, B.; Faria, D.; Couto, F.M.; Cruz, I.F.; Pesquita, C. OAEI 2020 Results for AML and AMLC. In Proceedings of the Workshop on Ontology Matching, CEUR, Athens, Greece, 2–6 November 2020; Volume 2788, pp. 154–160.
41. Pour, N.; Algergawy, A.; Amini, R.; Faria, D.; Fundulaki, I.; Harrow, I.; Hertling, S.; Jimenez-Ruiz, E.; Jonquet, C.; Karam, N.; et al. Results of the Ontology Alignment Evaluation Initiative 2020. In Proceedings of the Workshop on Ontology Matching, CEUR, Athens, Greece, 2–6 November 2020; Volume 2788, pp. 92–138.
42. Goy, A.; Accornero, C.; Astrologo, D.; Colla, D.; D'Ambrosio, M.; Damiano, R.; Leontino, M.; Lieto, A.; Loreto, F.; Magro, D.; et al. Fruitful Synergies between Computer Science, Historical Studies and Archives: The Experience in the PRiSMHA Project. In Proceedings of the International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management, KMIS, Vienna, Austria, 17–19 September 2019; Bernardino, J., Salgado, A., Filipe, J., Eds.; Scitepress: Setúbal, Portugal, 2019; Volume 3, pp. 225–230.
43. Gerstl, P.; Pribbenow, S. Midwinters. End Games, and Body Parts: A Classification of Part-whole Relations. *Int. J. Hum. Comput. Stud.* **1995**, *43*, 865–889. [CrossRef]
44. Tong, C.; Roberts, R.; Borgo, R.; Walton, S.; Laramee, R.S.; Wegba, K.; Lu, A.; Wang, Y.; Qu, H.; Luo, Q.; et al. Storytelling and visualization: An extended survey. *Information* **2018**, *9*, 65. [CrossRef]