Bernardo Pereira Nunes

Towards a well-interlinked Web through matching and interlinking approaches

TESE DE DOUTORADO

DEPARTAMENTO DE INFORMÁTICA
Programa de Pós-Graduação em Informática

Rio de Janeiro
February 2014

Pontifícia Universidade Católica
do Rio de Janeiro

Bernardo Pereira Nunes

Towards a well-interlinked Web through matching and interlinking approaches

TESE DE DOUTORADO

Thesis presented to the Programa de Pós-Graduação em Informática of the Departamento de Informática, PUC–Rio as partial fulfillment of the requirements for the degree of Doutor em Informática

Advisor: Prof. Marco Antonio Casanova
Co–Advisor: Prof. Wolfgang Nejdl

Rio de Janeiro
February 2014

Pontifícia Universidade Católica
do Rio de Janeiro

Bernardo Pereira Nunes

Towards a well-interlinked Web through matching and interlinking approaches

Thesis presented to the Programa de Pós-Graduação em Informática, of the Departamento de Informática do Centro Técnico Científico da PUC–Rio, as partial fulfillment of the requirements for the degree of Doutor.

Prof. Marco Antonio Casanova
Advisor
Departamento de Informática — PUC–Rio

Prof. Wolfgang Nejdl
Co–Advisor
L3S Research Center — Leibniz Universität Hannover

Prof. Antonio Luz Furtado
Departamento de Informática — PUC–Rio

Prof. Giseli Rabello Lopes
Departamento de Informática — PUC–Rio

Prof. Luiz André P. P. Leme
Instituto de Computação — UFF

Prof. Sean Wolfgand Matsui Siqueira
Departamento de Informática Aplicada — UNIRIO

Prof. Stefan Dietze
L3S Research Center — Leibniz Universität Hannover

Prof. José Eugenio Leal
Coordinator of the Centro Técnico Científico da PUC–Rio

Rio de Janeiro, February 10th, 2014

Bernardo Pereira Nunes

Bernardo Pereira Nunes holds a master in computer science, post–graduate in pedagogical mediation for distance education and a computer engineering degrees from Pontifical Catholic University of Rio de Janeiro (PUC–Rio). Currently, Bernardo is a senior system analyst of the Central Coordination for Distance Learning (a.k.a. CCEAD) of PUC–Rio. At CCEAD PUC–Rio, he has participated and led many academic and industry-oriented projects that have been awarded by Brazilian and International institutions. In addition, he has also worked as a research assistant at the L3S Research Center of Leibniz Universität Hannover in Germany, where he participated in Linked Data, Semantic Web, Data Mining and Web Archiving E.U. projects. His main research interest areas include Semantic Web, e-Learning, Information Retrieval, Information Extraction and Natural Language Processing. In these fields, Bernardo has published over 30 peer-reviewed scientific publications and Web applications as the well-known and broadcasted FireMe and Cite4Me.

Ficha Catalográfica

Pereira Nunes, Bernardo

Towards a well-interlinked Web through matching and interlinking approaches / Bernardo Pereira Nunes; advisor: Marco Antonio Casanova; co–advisor: Wolfgang Nejdl. — 2014.

88 f. : il. (color); 30 cm

1. Tese (doutorado) - Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Informática, 2014.

Inclui bibliografia.

1. Informática – Teses. 2. Integração de dados. 3. Consolidação de dados. 4. Linked data. 5. Web semântica. 6. Alinhamento de ontologias. 7. Alinhamento de esquemas. 8. Entity linking. 9. Document linking. 10. Sistemas de recomendação. 11. Cite4Me. 12. Privacidade. 13. FireMe. I. Casanova, Marco Antonio. II. Nejdl, Wolfgang. III. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Informática. IV. Título.

CDD: 004

This thesis is dedicated to the most important people in my life:
my mother, my brothers and my fiancée.

Acknowledgments

First, and foremost, I would like to express my sincerest gratitude to my advisor, Prof. Dr. Marco Antonio Casanova, who has supported me since my final project in computer engineering until today. Throughout these years, he devoted his time, patience, and unselfishly shared his knowledge and experience with me. There was no time zone or plane boarding that would stop him to answer my desesperate calls from Germany to a last minute advice before a paper submission. Many thanks for showing me as a researcher and supervisor should be through actions.

I would also like to offer my deepest gratitude to Dr. Stefan Dietze. His mentoring, enthusiasm, knowledge and friendship helped me to constantly improve and engage with my research. Without a shadow of doubt, you made my stay and my research in Germany more enjoyable with your friendship and dedication. Thanks for all the opportunities and support, which have definitely contributed to my personal and professional growth.

I would like to extend my appreciation and gratitude to Prof. Dr. Wolfgang Nejdl, who has always supported my stay at L3S Research Center. Since my first day at L3S, he oversaw and pointed out new directions in my research that helped me to improve and build a strong structure to this research. Thank you for your guidance and the opportunity to join the L3S team, this was a determinant step in my life.

I am also grateful to Prof. Dr. Antonio Luz Furtado. His maddening attention to detail and brilliant advice in my thesis proposal were of outmost value, it has been a real privilege to count on his creative support and guidance since my master and computer engineering final project, which has motivated me to start my PhD.

Thanks to all members of CCEAD PUC–Rio for being so supportive. I would like to dedicate a special thank you to Prof. Dr. Gilda Helena Bernardino de Campos, who unconditionally helped and supported me during my studies. Her support has been crucial to my professional and academic growth. Thanks for all opportunities, advice and to make my experience abroad possible.

I am also thankful to the immeasurable friendship of Besnik Fetahu and Ricardo Kawase, who were always willing to help and collaborate. Thanks for always being there challenging constructively and acting as mental sparring partners whenever needed and never abandon a friend to burn the midnight oil near to conference deadlines.

In the same way, I would like to thank Igor de O. Martins and Luiz Galvão for their unvaluable friendship. Their support while I was abroad was fundamental to help me focus only on my research. Thanks for solving all problems and paperwork, it kept my mind free to concentrate on my thesis.

Thanks to CAPES and PUC-Rio for the scholarships that enabled me to pursue this degree.

Last but not least, I am forever indebted with my mother, brothers and fiancée for all their support. This work would not have been possible without their endless comprehension, motivation, constant dedication, inexhaustible patience and heartfelt love. Thanks for always supporting my decisions, making my dreams come true and turning my life happier. All my achievements are yours. I love you all, thanks.

Abstract

Pereira Nunes, Bernardo; Casanova, Marco Antonio; Nejdl, Wolfgang. **Towards a well-interlinked Web through matching and interlinking approaches**. Rio de Janeiro, 2014. 88p. DSc Thesis — Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

With the emergence of Linked (Open) Data, a number of novel and notable research challenges have been raised. The "openness" that often characterises Linked Data offers an opportunity to homogeneously integrate and connect heterogeneous data sources on the Web. As disparate data sources with overlapping or related resources are provided by different data publishers, their integration and consolidation becomes a real challenge. An additional challenge of Linked Data lies in the creation of a well-interlinked graph of Web data. Identifying and linking not only identical Web resources, but also lateral Web resources, provides the data consumer with richer representation of the data and the possibility of exploiting connected resources. In this thesis, we present three approaches that tackle data integration, consolidation and linkage problems. Our first approach combines mutual information and genetic programming techniques for complex datatype property matching, a rarely addressed problem in the literature. In the second and third approaches, we adopt and extend a measure from social network theory to address data consolidation and interlinking. Furthermore, we present a Web-based application named Cite4Me that provides a new perspective on search and retrieval of Linked Open Data sets, as well as the benefits of using our approaches. Finally, we validate our approaches through extensive evaluations using real-world datasets, reporting results that outperform state of the art approaches.

Keywords

Data Integration; Data Consolidation; Linked Data; Semantic Web; Ontology Matching; Schema Matching; Entity Linking; Document Linking; Recommender Systems; Cite4Me; Privacy; FireMe.

Resumo

Pereira Nunes, Bernardo; Casanova, Marco Antonio; Nejdl, Wolfgang. **Interligando recursos na Web através de abordagens de matching e interlinking**. Rio de Janeiro, 2014. 88p. Tese de Doutorado — Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

Com o surgimento da Linked (Open) Data, uma série de novos e importantes desafios de pesquisa vieram à tona. A "abertura de dados", como muitas vezes a Linked Data é conhecida, oferece uma oportunidade para integrar e conectar, de forma homogênea, fontes de dados heterogêneas na Web. Como diferentes fontes de dados, com recursos em comum ou relacionados, são publicados por diferentes editores, a sua integração e consolidação torna-se um verdadeiro desafio. Outro desafio advindo da Linked Data está na criação de um grafo denso de dados na Web. Com isso, a identificação e interligação, não só de recursos idênticos, mas também dos recursos relacionadas na Web, provê ao consumidor (data consumer) uma representação mais rica dos dados e a possibilidade de exploração dos recursos conectados. Nesta tese, apresentamos três abordagens para enfrentar os problemas de integração, consolidação e interligação de dados. Nossa primeira abordagem combina técnicas de informação mútua e programação genética para solucionar o problema de alinhamento complexo entre fontes de dados, um problema raramente abordado na literatura. Na segunda e terceira abordagens, adotamos e ampliamos uma métrica utilizada em teoria de redes sociais para enfrentar o problema de consolidação e interligação de dados. Além disso, apresentamos um aplicativo Web chamado Cite4Me que fornece uma nova perspectiva sobre a pesquisa e recuperação de conjuntos de Linked Open Data, bem como os benefícios da utilização de nossas abordagens. Por fim, uma série de experimentos utilizando conjuntos de dados reais demonstram que as nossas abordagens superam abordagens consideradas como estado da arte.

Palavras–chave

Integração de dados; Consolidação de dados; Linked data; Web semântica; Alinhamento de ontologias; Alinhamento de esquemas; Entity linking; Document linking; Sistemas de recomendação; Cite4Me; Privacidade; FireMe.

Contents

List of figures

List of tables

1

Introduction

1.1 Motivation and Challenges

Due to the decentralised nature and the rapid growing of the World Wide Web, a large number of proprietary and competing terminologies and ontologies have been created describing similar or overlapping domains [1].

An example of equivalent or overlapping ontologies is *vCard*[1], *FOAF*[2] and *ORG Ontology*[3]. Although each ontology individually provides useful information, they essentially cover the same domain (i.e. people and organisations). Therefore, matching and reusing the overlapped parts of the competing ontologies would facilitate and enhance the access of the information on the Web.

Recently, the Linked (Open) Data paradigm and semantic Web technologies have emerged as an opportunity to homogeneously integrate and connect heterogeneous data sources to tackle interoperabilities issues between disparate (semantic) applications. A key challenge in providing interoperability is finding semantic correspondences between ontologies' elements, i.e. the *ontology matching* problem.

A fundamental component of ontology matching is *property mapping*, and yet there is little support that goes beyond the identification of single property matches amongst ontologies. For instance, real data often requires some degree of composition, trivially exemplified by the mapping of "first name" and "last name" to "full name" on one end, to complex matchings, such as parsing and pairing symbol/digit strings to SSN numbers, at the other end of the spectrum.

An additional challenge in the provision of a well-interlinked graph of Web

[1] http://www.w3.org/TR/vcard-rdf/
[2] http://xmlns.com/foaf/spec/
[3] http://www.w3.org/TR/vocab-org/

data lies in the identification and linkage of not only identical Web resources, but also lateral Web resources. The linkage of Web resources provides data consumers with a richer representation of the data and the possibility of exploiting and uncovering information by traversing the Web of Data graph.

Current interlinking techniques usually resort to mapping entities which refer to the same resource or real-world entity, e.g., by creating `owl:sameAs` references between an extracted entity representing the city "Berlin" with the corresponding Freebase[4] and Geonames[5] entries.

While relations within particular datasets are often well-defined, links between disparate datasets and corpora of Web resources are rare. The detection of *related* entities within and across datasets is of paramount importance. For instance, `skos:related` or `so:related` references can be created between entities that are to some degree connected [2, 3]. Moreover, latent connections can further be used as a means to unveil relationships between documents.

1.2 Contributions

This thesis reports contributions to data integration, consolidation and linkage problems, focusing on specific research challenges towards a well-interlinked Web.

Our first contribution to the Web of Data is the creation of a two-phase algorithm for complex datatype property matching, a rarely addressed problem in literature. Briefly, phase 1 of our approach computes the Estimated Mutual Information matrix of the property values to (1) find simple, 1:1 matches, and (2) compute a list of possible complex matches. Phase 2 applies Genetic Programming to the much reduced search space of candidate matches to find complex matches.

This contribution is validated through an extensive experimental evaluation using real-world data. The results show that the technique is able to find matches between sets of datatype properties with high accuracy, and that the proposed technique greatly improves results over those obtained if the Estimate Mutual Information matrix or the Genetic Programming techniques were to be used independently.

The second and third contributions address data consolidation and interlinking problems. These problems are tackled by means of connectivity of Web resources on the Web. Although relatedness is liable to subjective interpretations, connectivity is not. Given the Semantic Web's ability of linking Web resources, connectivity

[4] `http://www.freebase.com`
[5] `http://www.geonames.org`

can be measured by exploiting the links between entities and subsequently these connections can be exploited to uncover relationships between Web resources.

Thus, we present a combined approach to uncover relationships between disparate entities which exploits (a) graph analysis of reference datasets together with (b) entity co-occurrence on the Web with the help of search engines. In (a), we introduce a novel approach, adopted and applied from social network theory, to measure the connectivity between given entities in reference datasets. Furthermore, we expand and extend the connectivity measure between entities to measure and identify connected Web resources on the Web.

We validate our approaches through extensive evaluations using real-world datasets, reporting results that outperform state of the art approaches.

Furthermore, as a resulting contribution of this thesis, we introduce a Web-based application named *Cite4Me* that provides a new perspective on search and retrieval of Linked Open Data sets, as well as the benefits of using our approaches.

Cite4Me aims at providing a single access point for accessing papers and, therefore, assisting searchers on finding relevant papers, and unveiling new nomenclature more efficiently. For this, we use reference datasets, such as DBpedia, to explore semantic relationships between scientific papers and user queries.

As a final contribution, we briefly introduce *FireMe app*, a Web-based application used to show further directions in the present study and alert users of the impact of having data publicly available on the Web.

1.3 Impact

This thesis consists of papers that have been published in conferences of high relevance and prestige in the areas of Semantic Web, Web Science and Databases.

Our study regarding ontology matching as a possible solution to the problem of finding semantic mappings between datatype properties was initially published in [4]. It was subsequently expanded on and published in [5].

We also conducted studies in the area of entity and document interlinking. The work regarding the discovery of relationships between entities was originally published in [6] and, after an extensive evaluation, successfully resubmitted for publication in [7]. The study conducted to establish connections between related documents was published in [8].

A novel Web-based application built on previous approaches for exploratory search, retrieval and visualization of scientific publications was first published and awarded in [9]. After developing new functionalities, it was subsequently published in [10].

Another aspect of study present in this thesis was the examination of the impact and implications, which these interlinking approaches have regarding users' privacy. This work was published in [11] and received a lot of attention from the press (see Appendix A).

In the course of this research process, several other contributions were published in conferences and journals in related fields. The publications are [12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28].

1.4 Thesis Outline

The remainder of this thesis is structured as follows:

Chapter 2 presents a two-phase instance-based technique for complex datatype property matching.

Chapter 3 introduces a general-purpose approach to detect and measure semantic connectivity between entities within reference datasets.

Chapter 4 defines a measure to compute connectivity between documents in disparate datasets and document corpora.

Chapter 5 describes a Web application for exploratory search, retrieval and visualization of scientific publications based on the approaches described in the previous chapters.

Chapter 6 concludes with a summary of the contributions of the thesis, implications of our interlinking approaches and directions for future work.

2

Complex Datatype Property Matching

2.1 Introduction

Ontology matching is a fundamental problem in many applications areas [29]. Using OWL concepts, *by datatype property matching* we mean the special case of matching datatype properties from two classes.

Concisely, an *instance* of a datatype property p is a triple of the form (s, p, l), where s is a resource identifier and l is a literal. A *datatype property matching* from a *source class* S to a *target class* T is a partial relation μ between sets of datatype properties from S and T, respectively. We say that a match ***(A,B)*** $\in \mu$ is m:n *iff* ***A*** and ***B*** contain m and n properties, respectively. A match ***(A,B)*** $\in \mu$ should be accompanied by one or more *datatype property mappings* that indicate how to construct instances of the properties in ***B*** from instances of the properties in ***A***. A match ***(A,B)*** $\in \mu$ is *simple iff* it is 1:1 and the mapping is an identity function; otherwise, it is complex.

We introduce a two-phase, instance-based datatype property matching technique that is able to find complex n:1 datatype property matches and to construct the corresponding property mappings. The technique extends the ontology matching process described in [30] to include complex matches between sets of datatype properties and is classified as instance-based since it depends on sets of instances.

Thus, given two sets, s and t, that contain instances of the datatype properties of the source class S and the target class T, respectively, the first phase of the technique constructs the Estimated Mutual Information matrix (EMI) [30, 31] of the datatype property instances in s and t, which intuitively measures the amount of

related information from the observed property instances. The scope of this phase is to identify simple datatype property matches. For example, it may detect that the "e-mail" datatype property of one class matches the "electronic address" datatype property of another class. Additionally, the first phase suggests, for the second phase, sets of candidate datatype properties that can be matched only under more complex relationships, thereby reducing the search space.

The second phase uses a Genetic Programming approach (GP) to find complex n:1 datatype property matches. For example, it discovers that the "first name" and "last name" datatype properties of the source class match the "full name" datatype property of the target class, and returns a property mapping function that concatenates the values of "first name" and "last name" (of the same class instance) to generate the "full name" value. The reason for adopting genetic programming is two-fold:

1. It reduces the cost of traversing the search space;
2. It can be used to automatically generate complex mappings between datatype property sets.

The difficulty of the problem of finding complex matches between sets of datatype properties should not be underestimated since the search space is typically quite large. Therefore, our contribution towards a more accurate and efficient solution lies in proposing a two-phase technique, which deals with the problem of finding complex matches by:

(a) Using the estimated mutual information matrix (in Phase 1) as a preprocessing stage, limiting the candidate sets of properties for complex matches;

(b) Adopting a genetic programming strategy to automatically generate complex property mappings.

We also give empirical evidence that the combination of both approaches, estimated mutual information and genetic programming, yields better results than using either technique in separate. Furthermore, an example using real-world data is provided to convey the usefullness of our approach and describe it in details.

The remainder of this chapter is structured as follows. Section 2.2 summarises basic concepts that we use throughout the sections, while Section 2.3 presents our two-phase approach. Section 2.4 contains an example implementation of the technique. Section 2.5 and Section 2.6 present the evaluation setup and the results of our approach, respectively. Finally, Section 2.7 outlines related literature and Section 2.8 presents the conclusions.

2.2 Background

2.2.1 Vocabulary Matching and Concept Mapping

We decompose the problem of OWL ontology matching into the problem of vocabulary matching and that of concept mapping. In this section, we briefly review these concepts and extend them to account for complex property matching. In what follows, let ***S*** and ***T*** be two OWL ontologies, and $\boldsymbol{V_S}$ and $\boldsymbol{V_T}$ be their vocabularies, respectively. Let $\boldsymbol{C_S}$ and $\boldsymbol{C_T}$ be the sets of classes and $\boldsymbol{P_S}$ and $\boldsymbol{P_T}$ be the sets of properties in $\boldsymbol{V_S}$ and $\boldsymbol{V_T}$, respectively.

An *instance* of a class c is a triple of the form $(s, rdf{:}type, c)$, an instance of an object property p is a triple of the form (s, p, o) and an instance of a datatype property d is a triple of the form (s, d, l), where s and o are resource identifiers and l is a literal.

A *vocabulary matching* between ***S*** and ***T*** is a finite set $\mu \subseteq \boldsymbol{V_S} \times \boldsymbol{V_T}$. Given $(v_1, v_2) \in \mu$, we say that (v_1, v_2) is a *match* in μ and that μ *matches* v_1 with v_2; a *property* (or *class*) *matching* is a matching defined only for properties (or classes).

A *concept mapping* from ***S*** to ***T*** is a set of transformation rules that map instances of the concepts of ***S*** into instances of the concepts of ***T***.

Here, we extend *vocabulary matchings* to also include pairs of the form ***(A,B)*** where ***A*** and ***B*** are sets of datatype properties in $\boldsymbol{P_S}$ and $\boldsymbol{P_T}$, respectively. We say that ***(A,B)*** is an $m{:}n$ match *iff* ***A*** and ***B*** contain m and n properties, respectively. In this case, a match ***(A,B)*** must be accompanied by *datatype property mappings*, denoted $\mu[\boldsymbol{A}, B_i]$, which are transformation rules that map instances of the properties in ***A*** into instances of the property B_i, for $i = 1, \ldots, n$, where $\boldsymbol{B}=\{B_1, \ldots, B_n\}$. Using "//" to denote string concatenation, the following transformation rule:

$$(s, fullName, v) \leftarrow (s, firstName, n), (s, lastName, f), v = n//f$$

indicates that the value of the "fullName" property is obtained by concatenating the values of properties "firstName" and "lastName". We will use the following abbreviated form for mapping rules with the above syntax:

$$\mu[\{firstName, lastName\}, fullName] =$$

$$\text{“}fullName \leftarrow firstName//lastName''$$

As an abuse of notation, when ***A*** is a singleton $\{A_1\}$, we simply write $\mu[A_1, B_i]$,

rather than $\mu[\{A_1\}, B_i]$. Finally, a match ***(A,B)*** is *simple iff* it is 1:1, that is, of the form $(\{A_1\}, B_1)$, and the mapping $\mu[A_1, B_1]$ is the *identity transformation rule*, defined as "$(s, B_1, l) \leftarrow (s, A_1, l)$"; otherwise, the match is *complex*.

2.2.2 An Instance-Based Process for Vocabulary Matching

In this section, we very briefly summarise the instance-based process to create *vocabulary matchings* introduced in [30]. The outline of the process is as follows:

S1. Generate a preliminary property matching using similarity functions;

S2. Generate a class matching using the property matching obtained in S1;

S3. Generate an instance matching using the output from S1;

S4. Refine the property matching using the class matching generated in S2 and the instance matching from S3.

The final vocabulary matching is the result of the union of the class matching obtained in S2 and the refined property matching obtained in S4.

The intuition used in all steps of the process is that *"two schema elements match iff they have many values in common and few values not in common"*, i.e. *iff* they are similar above a given similarity threshold.

We obtain the following output from each individual step. S1 generates preliminary 1:1 property matchings based on the intuition that two properties match *iff* their instances share similar sets of values. In the case of string properties, their values are replaced by the tokens extracted from their values. S1 provides evidences on class and instance matchings, explored in the next two steps.

S2 generates class matchings based on the intuition that two classes match *iff* their sets of properties are similar. This step uses the property matchings generated in S1.

S3 generates instance matchings based on the intuition that two instances match *iff* the values of their properties are similar. However, equivalent instances from different classes may be described by very different sets of properties. Therefore, extracting values from all of their properties may lead to the wrong conclusion that the instances are not equivalent. Therefore, Leme et al. [30] propose to extract values only from the matching properties of the instances.

2.3 Two-Phase Property Matching Technique

In this section, we introduce a technique to partly implement and extend the ontology matching process of Section 2.2.2 to compute complex n:1 datatype property matches (note that the technique does not cover n:m matches). The technique comprises two phases:

- **Phase 1** uses Estimated Mutual Information matrices, defined further in Section 2.3.1, to compute 1:1 simple matches;
- **Phase 2** uses genetic programming to compute complex n:1 matches, based on the information outputted from Phase 1.

2.3.1 Phase 1: Computing Simple Datatype Property Matches with Estimated Mutual Information

Let $\boldsymbol{p}$=(p_1,...,p_u) and $\boldsymbol{q}$=(q_1,...,q_v) be two lists of sets. The *co-occurrence matrix* of $\boldsymbol{p}$ and $\boldsymbol{q}$ is defined as the matrix $[m_{ij}]$ such that $m_{ij} = |p_i \cap q_j|$, for $i \in [1, u]$ and $j \in [1, v]$. The *Estimated Mutual Information matrix* of $\boldsymbol{p}$ and $\boldsymbol{q}$ is defined as the matrix $[EMI_{pq}]$ such that:

$$EMI_{pq} = \frac{m_{pq}}{M} \cdot log \left(M \cdot \frac{m_{pq}}{\sum_{j=1}^{v} m_{pj} \cdot \sum_{i=1}^{u} m_{iq}} \right) \quad (2\text{-}1)$$

where $M = \sum_{i=1}^{u} \sum_{j=1}^{v} m_{ij}$.

We now adapt these concepts to define Phase 1 of the datatype property matching process. Let S and T be two classes with sets of datatype properties $\boldsymbol{A}$=$\{A_1, \ldots, A_u\}$ and $\boldsymbol{B}$=$\{B_1, \ldots, B_v\}$, respectively. Let $\boldsymbol{s}$ and $\boldsymbol{t}$ be sets of instances of the properties in $\boldsymbol{A}$ and $\boldsymbol{B}$, respectively ($\boldsymbol{s}$ and $\boldsymbol{t}$ therefore are sets of RDF triples).

Rather than simply using the cardinality of set intersections to define the co-occurrence matrix $[m_{ij}]$, Phase 1 computes $[m_{ij}]$ using *set comparison functions* that take two sets and return a non-negative integer. Such functions play the role of *flexibilization points* of Phase 1, as illustrated in Section 2.4.1.

The set comparison functions depend on the types of the values of the datatype properties as well as on whether the functions take advantage of instance matches. For example, given a pair of datatype properties, A_i and B_j, m_{ij} may be defined as

the number of pairs of triples (a, A_i, b) in $\boldsymbol{s}$ and (c, B_j, d) in $\boldsymbol{t}$ such that instances a and c match (or are identical) and the literals b and d are equal (or are considered equal, under a *literal comparison function* defined for the specific datatype of b and d).

For instance, Leme et al. [30] adopt the cosine similarity function to compare strings. The cosine similarity is defined as:

$$\cos(x, y) = \frac{x \cdot y}{\| x \| \| y \|} \tag{2-2}$$

where x and y are the vectors of tokens obtained from the strings; m_{ij} is then computed as the number of (string) values of triples for property A_i in $\boldsymbol{s}$ whose cosine distance to values of instances for property B_j in $\boldsymbol{t}$ is above a given threshold ($\alpha \geq 0.8$ in [30]).

To compute simple matches (1:1), the cosine similarity function proved to be appropriate, especially if the strings to be compared have approximately the same number of tokens. However, the cosine similarity function turned out not to be appropriate when using the co-occurrence matrix to suggest complex matches to Phase 2 of the technique. We therefore adopted the Jaccard similarity coefficient to compute the co-occurrence matrix, defined as

$$Jaccard(b, d) = \frac{|b \cap d|}{|b \cup d|} \tag{2-3}$$

which counts the number of tokens that strings b and d have in common.

Thus, given two properties A_i and B_j, m_{ij} is computed as the sum of $Jaccard(A_i, B_j)$, for all pairs of strings b and d such that there are triples of the form (a, A_i, b) in $\boldsymbol{s}$ and (c, B_j, d) in $\boldsymbol{t}$.

Phase 1 proceeds by computing the EMI matrix based on the co-occurrence matrix, as in Eq. 2-1. Next, it computes a 1:1 matching, μ_{EMI}, between the properties in $\boldsymbol{A}$=$\{A_1, \ldots, A_u\}$ and those in $\boldsymbol{B}$=$\{B_1, \ldots, B_v\}$ such that, for any pair of properties A_p and B_q, $(A_p, B_q) \in \mu_{EMI}$ *iff* $EMI_{pq} > 0$ and $EMI_{pj} \leq 0$, for all $j \in [1, v]$, with $j \neq q$, and $EMI_{iq} \leq 0$, for all $i \in [1, u]$, with $i \neq p$. Furthermore, Phase 1 assumes that the property mappings, $\mu_{EMI}[A_r, B_s]$, are always the identity function.

Finally, Phase 1 also outputs a list of datatype properties to be considered for complex matching in Phase 2. For the k^{th} column of the EMI matrix, it outputs the pair $(\boldsymbol{A^k}, B_k)$ as a candidate n:1 complex match, where B_k is the property of T that corresponds to the k_{th} column and $\boldsymbol{A^k}$ is the set of properties A_i of S such

that $EMI_{ik} > 0$. Indeed, if $EMI_{ik} \leq 0$, then A_i and B_k have no information in common. However, note that this heuristic does not indicate what is a candidate property mapping $\mu[\boldsymbol{A^k}, B_k]$. This problem is faced in Phase 2.

2.3.2 Phase 2: Computing Complex Property Matches with Genetic Programming

The second phase of the technique uses genetic programming to create mappings between the properties that have some degree of correlation, as identified in the first phase. Briefly, the process goes as follows.

Recall that *genetic programming* refers to an automated method to create and evolve programs to solve a problem [32]. A *program*, also called an *individual* or a *solution*, is represented by a tree, whose nodes are labeled with functions (concatenate, split, sum, etc) or with values (strings, numbers, etc). New individuals are generated by applying genetic operations to the current population of individuals.

Note that genetic programming does not enumerate all possible individuals, but it selects individuals that should be bred by an evolutionary process. The *fitness function* assigns a *fitness value* to each individual, which represents how close an individual is to the solution and determines the chance of the individual to remain in the genetic process. Algorithm 1 models the process of breeding new individuals.

Algorithm 1 Breeding new individuals.

Randomly generate an initial set of individuals I;

```
/* The stop criterion could be a threshold of the best-so-far
individual evaluated by the fitness function or the number of
generations of individuals.*/
```

while *stop criterion* **do**

  for *each individual* $\in$ *I* **do**

    IndividualScore = fitnessValue;

  end

  Create using the genetic operations (reproduction, mutation or crossover) a new set of individuals from the individuals chosen by the probability based on the fitness value.

end

```
/* This result may be a solution (or an approximate solution) to
the problem. */
```

Return the best-so-far individual;

The process requires two configuration steps, carried out just once. First, certain parameters of the process must be properly calibrated to prevent overfitting problems, to avoid unnecessary runtime overhead, and to help finding good solutions.

Concisely, the calibration process is performed using the 4-fold cross validation process, where the dataset used to calibrate the parameters is partitioned into four nearly equally sized partitions and four iterations of training and validation are performed. For each iteration, a different subset of the data is selected for validation, while the remaining three subsets are used for learning, thereby guaranteeing the selection of the most suitable genetic parameters configuration (see Table 2.1).

Table 2.1: Adjusted genetic parameters.

Parameter	Adjusted values
Population size ($\sigma_{population}$)	40
Maximum height (σ_{height})	3
Number of generations ($\sigma_{generations}$)	50
Mutation rate ($\sigma_{mutation}$)	2%
Crossover proportion ($\sigma_{crossover}$)	60%
Reproduction proportion ($\sigma_{reproduction}$)	40%

Once the parameters are calibrated, the second configuration step is to determine the stop criterion. We opted to stop after a predetermined maximum number of generations and return the best-so-far individual to limit the cost of searching for individuals.

We now show how to use genetic programming to compute complex datatype property matches. Let S and T be two classes with sets of datatype properties $\boldsymbol{A}$=$\{A_1, \ldots, A_u\}$ and $\boldsymbol{B}$=$\{B_1, \ldots, B_v\}$, respectively. Let $\boldsymbol{s}$ and $\boldsymbol{t}$ be lists of sets of instances of the properties in $\boldsymbol{A}$ and $\boldsymbol{B}$, respectively.

The genetic programming phase receives as input the candidate matches that Phase 1 outputs and the sets $\boldsymbol{s}$ and $\boldsymbol{t}$. For each input candidate match, it outputs a property mapping $\mu[\boldsymbol{A^k}, B_k]$, if one exists; otherwise it discards the candidate match.

Let $(\boldsymbol{A^k}, B_k)$ be a candidate match output by the first phase, where $\boldsymbol{A^k}$ is a set of properties in $\boldsymbol{A}$ and B_k is a property in $\boldsymbol{B}$. The genetic programming phase first generates a random initial population of candidate property mappings. In each iteration step, it creates new candidate property mappings using genetic operations. It keeps the best-so-far individual, and returns it when the stop criterion is reached.

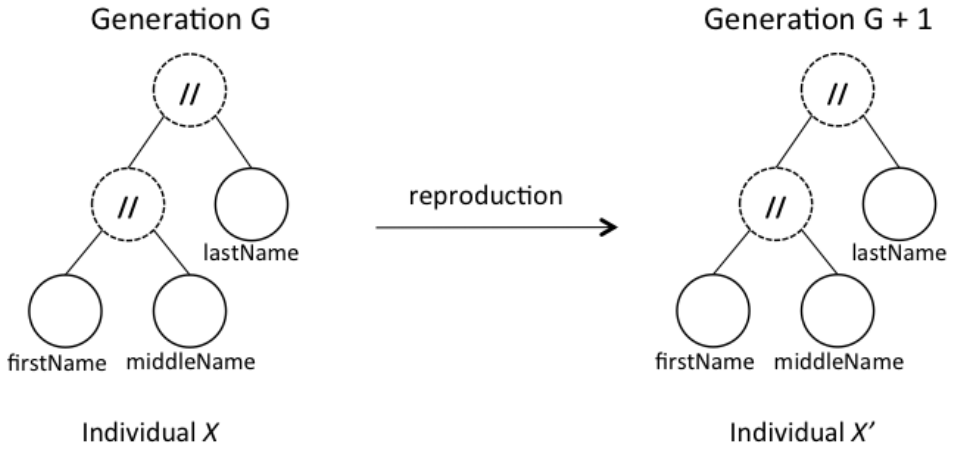


Figure 2.1: Example of the reproduction operation. An exact copy of an individual x is placed in the next generation to preserve good individuals.

Again, the process depends on the specifications presented in Table 2.1, which should be regarded as flexibilization points (as also presented in [33]).

A candidate property mapping $\mu[\boldsymbol{A^k}, B_k]$ (the individual in this case) is represented as a tree whose leaves are labeled with the properties in $\boldsymbol{A^k}$ and whose internal nodes are labeled with primitive mapping functions.

As described in Table 2.1, the maximum population size, $\sigma_{population}$, is a parameter of the process. The initial population consists of $\sigma_{population}$ randomly generated trees. Each tree has a maximum height, defined by the parameter σ_{height}, each leaf is labeled with a property from $\boldsymbol{A^k}$ and each internal node is labeled with a primitive mapping function.

The reproduction operation simply preserves a percentage of the property mappings from one generation to the next, defined by the parameter $\sigma_{reproduction}$. The reproduction reduces the risk of losing individuals that are best fitted to solve the problem. Figure 2.1 illustrates the operation.

The crossover operation exchanges subtrees of two candidate property mappings to create new candidate mappings. For example, suppose that $\boldsymbol{A^k}$=$\{firstName, middleName, lastName\}$ and B_k=$fullName$ and consider the following two candidate property mappings (which use the concatenation operation, "//", and are represented using the notation adopted in Section 2.2.1):

$$\mu_1[\boldsymbol{A^k}, B_k] = \text{“}fullName \leftarrow (lastName//\textbf{(firstName // middleName)})\text{”}$$

$$\mu_2[\boldsymbol{A^k}, B_k] = \text{“}fullName \leftarrow (\textbf{(middleName // firstName)}//lastName)\text{”}$$

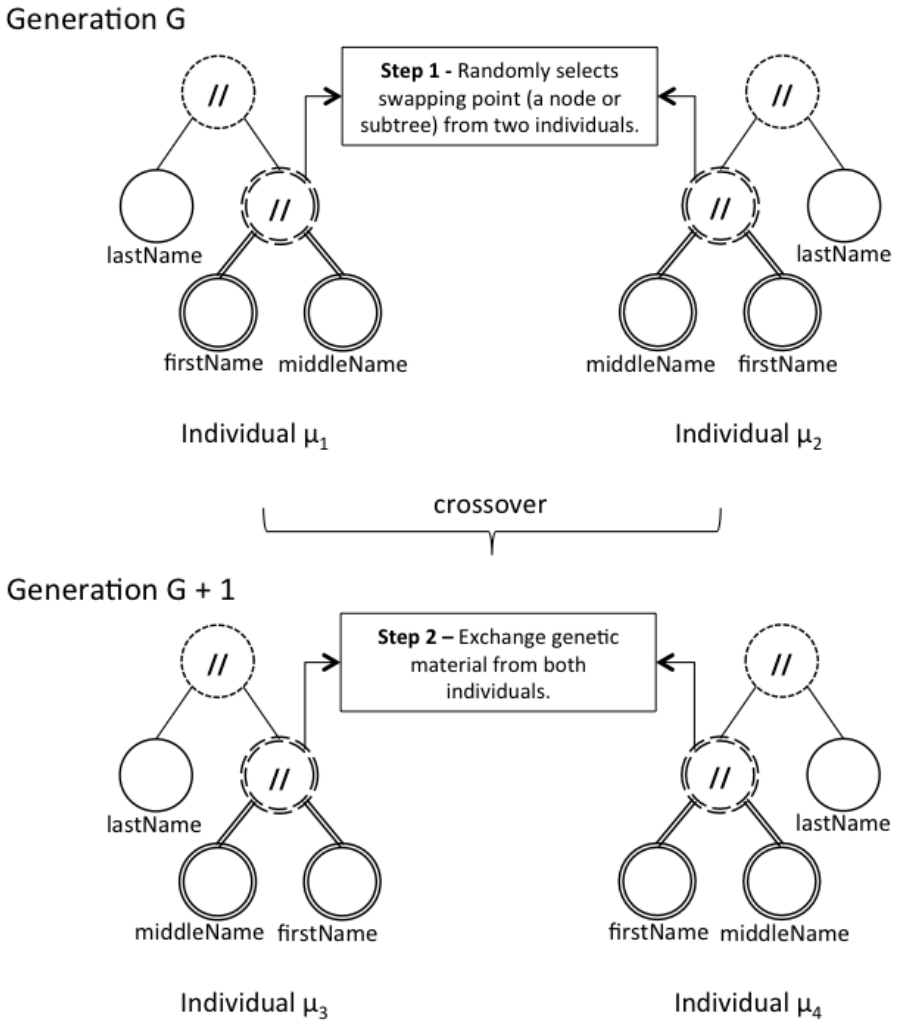


Figure 2.2: Example of the crossover operation. Step 1 randomly selects the swapping point of two individuals to generate new individuals with both genetic material in Step 2.

The crossover operation might generate the following two new candidate property mappings (by swapping the sub-expressions in boldface):

$$\mu_3[\boldsymbol{A^k}, B_k] = \text{“}fullName \leftarrow (lastName//\textbf{(middleName // firstName)})\text{”}$$

$$\mu_4[\boldsymbol{A^k}, B_k] = \text{“}fullName \leftarrow (\textbf{(firstName // middleName)}//lastName)\text{”}$$

Thus, the crossover operation increases the diversity of the population, while preserving some characteristics from the best individuals. Figure 2.2 illustrates the crossover operation.

The mutation operation randomly alters a node (labeled with a property or

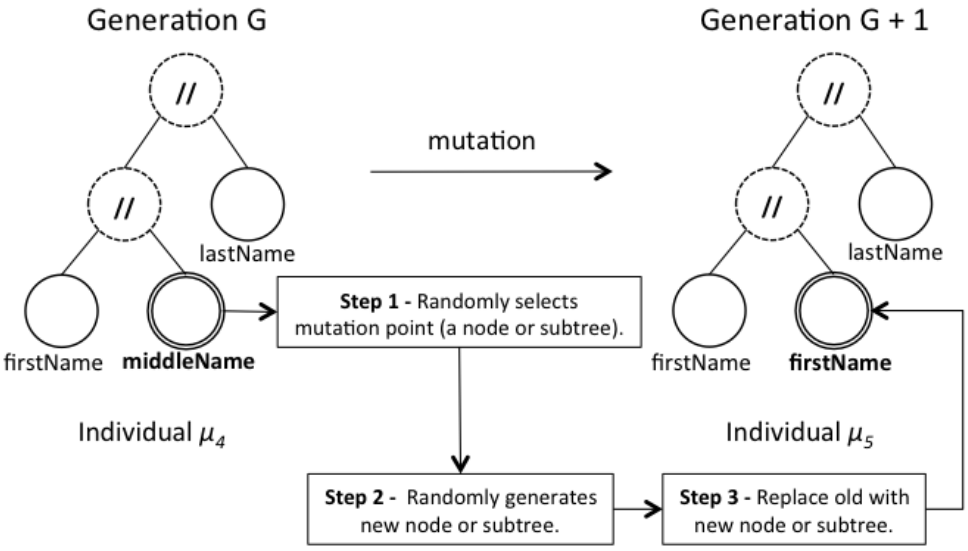


Figure 2.3: Example of the mutation operation. Step 1 randomly selects the mutation point of an individual. Step 2 randomly generates a new node or subtree. Finally, Step 3 replaces the old node/subtree with the new generated one.

with a primitive mapping function) of a candidate property mapping, introducing new *genetic material* to keep and increase population diversity.

For example, the node labeled with "middleName" of $\mu_4[\boldsymbol{A^k}, B_k]$ can be mutated to "firstName", resulting in a new candidate property mapping. Figure 2.3 shows an illustration example of the mutation operation. Note that, although this example is acceptable, it is not quite reasonable, since it repeats "firstName":

$$\mu_5[\boldsymbol{A^k}, B_k] = \text{“}fullName \leftarrow ((firstName//firstName)//lastName)\text{”}$$

Finally, recall that $\boldsymbol{s}$ and $\boldsymbol{t}$ are lists of sets of instances of the properties in $\boldsymbol{A}$ and $\boldsymbol{B}$, respectively. The fitness value of $\mu[\boldsymbol{A^k}, B_k]$ is computed by applying $\mu[\boldsymbol{A^k}, B_k]$ to the instances of the properties in $\boldsymbol{A^k}$ occurring in $\boldsymbol{s}$, creating a new set of instances for B_k, which is then compared with the set of instances of B_k occurring in $\boldsymbol{t}$. As in Section 2.3.1, the exact nature of the fitness function depends on the types of the values of the datatype properties as well as on whether the function takes advantage of instance matches or not (which is possible when implementing S4, see Section 2.2.2). For instance, we used the Levenshtein similarity function for string values and KL-divergence measure [34] for numeric values.

The Levenshtein similarity function is normalised to fall into the interval $[0, 1]$, where "1" indicates that a string is exactly equal to the other and "0" that the

two strings have nothing in common, while the KL-divergence measure is used to compute the similarity between two value distributions.

Recall that we are given two samples, $\boldsymbol{p}$ and $\boldsymbol{q}$, of instances of properties of classes P and Q, respectively. Construct the set X of strings that occur as literals of instances of B_k obtained by applying $\mu[\boldsymbol{A^k}, B_k]$ to $\boldsymbol{p}$, and the set Y of strings that occur as literals of instances of B_k in $\boldsymbol{q}$. The fitness score for a candidate property mapping is:

$$Fitness_{string}(\mu[\boldsymbol{A^k}, B_k]) = \frac{1}{n} \sum_{\substack{x \in X \\ y \in Y}} Levenshtein(x, y) \tag{2-4}$$

where n is the number of pairs in $X \times Y$.

In the case of numeric values, construct the set X of numeric values that occur as literals of instances of B_k, obtained by applying $\mu[\boldsymbol{A^k}, B_k]$ to $\boldsymbol{p}$, and the set Y of numeric values that occur as literals of instances of B_k in $\boldsymbol{q}$. The fitness score for a candidate property mapping is:

$$Fitness_{numeric}(F, G) = \frac{1}{n} \sum_{\substack{x \in X \\ y \in Y}} ln \left(\frac{F(x)}{G(y)} \right) F(x) \tag{2-5}$$

where n is the number of pairs in $X \times Y$, $F(x)$ represents the target distribution of instances in X and $G(y)$ is the the set of materialised mapping μ in Y from the source distribution of instances.

2.4 An Example Implementation

With the help of an example, we illustrate how to implement the two-phase technique. We assume that the implementation is in the context of S1 of the process described in Section 2.2.2, that is, we will not use instance matches. We start with Phase 1, described in Section 2.3.1.

The example is based on personal information classes, modeled by class P, with 7 properties and class Q with 3 properties. Table 2.2 shows the properties from the two classes P and Q, and also indicates which properties or sets of properties match. For example, $\{A_1, A_2\}$ matches B_1.

Table 2.2: Example schemas.

#	P	#	Q
A_1	FirstName	B_1	FullName (FirstName // LastName)
A_2	LastName		
A_3	E-Mail	B_2	E-Mail
A_4	Address	B_3	FullAddress (Address // Number // Complement // Neighborhood)
A_5	Number		
A_6	Complement		
A_7	Neighborhood		

2.4.1 Phase 1: Computing Simple Property Matches with Estimated Mutual Information

Recall from Section 2.3.1 that an implementation of Phase 1 requires defining set comparison functions used to compute the co-occurrence matrix $[m_{ij}]$. We discuss this point in what follows, with the help of the running example.

We assume that all property values are string literals and that we are given two samples, $\boldsymbol{p}$ and $\boldsymbol{q}$, of instances of properties of classes P and Q, respectively (each with 500 instances).

As mentioned in Section 2.3, Leme et al. [30] use the cosine similarity function to compute the co-occurrence matrix, which is able to indicate only simple 1:1 matches. By contrast, we used the Jaccard similarity coefficient that measures the similarity between sets, which is able to find simple 1:1 matches and *suggest* complex matches.

Figure 2.4 (a) shows the co-occurrence matrix computed using the cosine similarity measure. Note that $m_{43} = 164k$, which is high because the values of A_4 and B_3 come from a controlled vocabulary with a small number of terms (not indicated in Table 2.2). By contrast, $m_{32} = 500$, which is low because A_3 and B_2 are keys (also not indicated in Table 2.2).

Figure 2.4 (b) shows the co-occurrence matrix computed using the Jaccard similarity (see Eq. 2-3), which measures the similarity and diversity between sets. Thus, the co-occurrence indices are more sparse between the attributes that have values in common.

To clarify, consider A_7 (Neighborhood) and B_3 (FullAddress) and suppose that "Cambridge" is an observed value of A_7 and "* Oxford Street Cambridge MA, United States" of B_3. The cosine similarity of these two strings is 0.37, which is lower than the threshold set by [30] (again, $\alpha \geq 0.8$). Hence, these two strings

are considered not to be similar. However, also observe that “Cambridge” is fully contained in “* Oxford Street Cambridge MA, United States”, which might indicate that A_7, perhaps concatenated with the values of other datatype properties, might match B_3. Continuing with this argument, lowering the threshold also proved not to be efficient to account for these situations, since this increases noise in the matching process.

Thus, given two properties A_i and B_j, m_{ij} is computed as the sum of $Jaccard(x, y)$, for all pairs of strings x and y such that there are triples of the form (a, A_i, x) in $\boldsymbol{p}$ and (b, B_j, y) in $\boldsymbol{q}$ (see Figure 2.4). Once the co-occurrence matrix $[m_{ij}]$ is obtained, we compute the EMI matrix $[EMI_{ij}]$, as described in Section 2.3.1 (see Figure 2.5).

The result of Phase 1 therefore is the matching μ_{EMI} between the sets of properties $\{A_1, \ldots, A_u\}$ and $\{B_1, \ldots, B_v\}$, computed as in Section 2.3.1 (which we recall is $1:1$), assuming that, for each $(A_i, B_j) \in \mu_{EMI}$, the property mappings $\mu[A_i, B_j]$ is always the identity function (see Figure 2.5).

$$\begin{array}{c} \\ A_1 \\ A_2 \\ A_3 \\ A_4 \\ A_5 \\ A_6 \\ A_7 \end{array} \begin{array}{c} \begin{array}{ccc} B_1 & B_2 & B_3 \end{array} \\ \left[\begin{array}{ccc} 4 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 500 & 0 \\ 0 & 0 & 164k \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{array}\right] \end{array}$$

(a)

$$\begin{array}{c} \\ A_1 \\ A_2 \\ A_3 \\ A_4 \\ A_5 \\ A_6 \\ A_7 \end{array} \begin{array}{c} \begin{array}{ccc} B_1 & B_2 & B_3 \end{array} \\ \left[\begin{array}{ccc} 4,8k & 0 & 1,6k \\ 12,3k & 0 & 5,1k \\ 0 & 500 & 0 \\ 5,5k & 0 & 55k \\ 0 & 0 & 726 \\ 797 & 0 & 8,5k \\ 750 & 0 & 9,5 \end{array}\right] \end{array}$$

(b)

Figure 2.4: Co-occurrence matrices using (a) cosine similarity and (b) Jaccard similarity coefficient.

$$\begin{array}{c} \\ A_1 \\ A_2 \\ A_3 \\ A_4 \\ A_5 \\ A_6 \\ A_7 \end{array} \begin{array}{c} \begin{array}{ccc} B_1 & B_2 & B_3 \end{array} \\ \left[\begin{array}{ccc} 0{,}0550 & 0,0 & 0{,}0040 \\ 0{,}0138 & 0,0 & 0{,}0020 \\ 0,0 & 0{,}0020 & 0,0 \\ 0,0 & 0,0 & 0{,}0677 \\ 0,0 & 0,0 & 0{,}0090 \\ 0{,}0024 & 0,0 & 0{,}0094 \\ 0{,}0002 & 0,0 & 0{,}0114 \end{array}\right] \end{array}$$

Figure 2.5: EMI matrix: dark gray cells represent simple matches and light gray cells represent possible complex matches for the property in the column.

2.4.2 Phase 2: Computing Complex Property Matches with Genetic Programming

The second phase of the technique was implemented using a genetic programming toolkit [35], with the parameters shown in Table 2.1.

The first phase of the technique outputs, for instance, a candidate match between properties A_1, A_2, A_4, A_5, A_6 and A_7 (FirstName, LastName, Address, Number, Complement and Neighborhood, respectively) and property B_3 (FullAddress), see Figure 2.5. Note that quite frequently streets are named after famous people, which justifies why EMI outputs A_1 and A_2 as candidates properties. Following the example, having 6 properties as input, the genetic process begins the search for the solution.

As the property values are strings, the fitness function selected to find the best individual is the Levenshtein (see Eq. 2-4). Thus, after randomly generate an initial set of individuals, the fitness function assigns to each individual a score. For each new generation, a new set of individuals is created from those individuals chosen according to a probability based on their fitness value. After a predetermined number of generations, the process stops with an expression that represents a property mapping that maps the concatenation of the properties A_4, A_5, A_6 and A_7, that is, the expression:

$$((Address//Number)//(Complement//Neighborhood))$$

into property B_3 (that is, FullAddress).

2.5 Evaluation Setup

2.5.1 Datasets

For this evaluation, we use three datasets from different domains, where each of them contains a source and target schemas, a list of mappings amongst the schemas and sample data. Table 2.3 lists and describes the datasets and their schema information. The “Personal Information” dataset lists information about people, the “Real Estate” dataset lists information about houses for sale, while the “Inventory” dataset describes product inventories.

The “Real Estate” and “Inventory” datasets were extracted from a well-known

repository[1] used to evaluate schema and ontology matching approaches, while the "Personal Information" dataset was provided by a Brazilian University to assist them in a system migration and data integration problem.

Note that the low number of instances available in the "Real Estate" and "Inventory" datasets is purposeful and makes the matching problem even more challenging, since most of the instance-based approaches are more likely to find similar instances amongst large datasets than in sampled data.

Table 2.3: Description of the datasets from different domains.

Datasets	Total #Instances	Type		Total #Mappings
Personal Information	6000	String	1:1	12
			1:n	5
		Numeric	1:1	0
			1:n	0
Inventory	100	String	1:1	4
			1:n	4
		Numeric	1:1	25
			1:n	4
Real Estate	100	String	1:1	6
			1:n	5
		Numeric	1:1	1
			1:n	3

2.5.2 Ground Truth

In order to create the ground truth, a team consisting of two specialists in data integration manually analysed and classified the datatype properties of each schema as "string" and "numeric" types which, in turn, were subsequently subclassified as simple (1:1) and complex (n:1) matches. As shown further in Section 2.6, some approaches deal better with "string" datatype properties and simple matches than with "numeric" datatype properties and complex matches. Thus, this classification is crucial to compare and contrast the strengths and weaknesses of each evaluated approach.

Finally, the specialists in data integration created 79 property mapping functions between the dataset schemas to serve as ground truth and measure the performance of the approaches under the same conditions. The total number of mappings that semantically models equivalent datatype properties for each dataset and amongst their schemas is shown in Table 2.3.

[1]With exception of the "Personal Information" dataset due to privacy reasons, other datasets are available at `http://pages.cs.wisc.edu/~anhai/wisc-si-archive/domains/`.

2.5.3 Evaluation Methods

The first evaluation performed is the comparison of the two approaches, Estimated Mutual Information and Genetic Programming, when separately applied. Next, as a baseline, we compare our method against two state of the art methods [36, 37].

Thus, we compared our approach with the iMap system [36], which similarly to our approach addresses the problem of 1:1 and n:1 (complex) matches. Briefly, iMap transforms the matching problem into a search problem and looks for mappings in a predefined list of functions.

Finally, we compare our approach against the Learning Source Description approach (LSD) [37], which is able to find simple 1:1 matches using a set of base learners to predict the mapping functions.

2.5.4 Evaluation Metrics

The performance of the matching approaches is measured using standard metrics of precision (P), recall (R) and $F1$ measure. They are computed based on *true positive* (TP), *false positive* (FP) and *false negative* (FN) indicators. Eq. 2-6 shows how the precision is computed.

$$P = \begin{cases} 0, \text{ iff } |TP + FP| = 0 \\ \\ \dfrac{TP}{TP + FP}, \text{ otherwise} \end{cases} \tag{2-6}$$

where TP is the number of property mapping functions correctly found and FP is the number of property mapping functions wrongly created by the approach.

As for the recall, it is defined as follows:

$$R = \begin{cases} 0, \text{ iff } |TP + FN| = 0 \\ \\ \dfrac{TP}{TP + FN}, \text{ otherwise} \end{cases} \tag{2-7}$$

where FN indicates the missed correct property mapping functions.

Finally, $F1$ measures the harmonic average between precision and recall (see Eq. 2-8).

Table 2.4: P/R/F1 results for three datasets in different domains.

	EMI			GP			EMI+GP		
Dataset	P	R	F1	P	R	F1	P	R	F1
Personal Information	1	0.38	0.54	0.8	0.75	0.77	**1**	**0.94**	**0.96**
Inventory	1	0.24	0.39	0.96	0.87	0.91	**0.97**	**0.97**	**0.97**
Real Estate	1	0.33	0.5	1	0.47	0.64	**1**	**0.8**	**0.89**

$$F1 = \begin{cases} 0, & \text{iff } |P + R| = 0 \\ 2 \cdot \dfrac{P \cdot R}{P + R}, & \text{otherwise} \end{cases} \tag{2-8}$$

2.6 Results

Table 2.5: Mapping results for EMI, GP and EMI+GP.

Datasets	Type		EMI	GP	**EMI+GP**	Total #Mappings
Personal Information	String	1:1	6	12	**12**	12
		1:n	11*	1	**4**	5
	Numeric	1:1	0	0	**0**	0
		1:n	0	0	**0**	0
Inventory	String	1:1	3	4	**4**	4
		1:n	18*	2	**4**	4
	Numeric	1:1	6	25	**25**	25
		1:n	18*	1	**3**	4
Real Estate	String	1:1	4	4	**6**	6
		1:n	7*	2	**5**	5
	Numeric	1:1	1	1	**1**	1
		1:n	7*	0	**0**	3

($*$) Complex matches suggested by EMI.

For the first part of the evaluation, we present the results for the estimated mutual information, the genetic programming and the combination of the approaches (EMI+GP) applied separately. Column "EMI" of Table 2.4 indicates that, using only the estimated mutual information approach, we obtained a precision of 1.0 for all datasets, which indicates that none of the matches were mistakenly found; the rate of recall was low, between 0.24 and 0.38, indicating a high rate of missed property matches; and the $F1$ measure varied from 0.39 to 0.54, hinting that this approach is insufficient to find simple and complex matches. Indeed, out of the 12 simple matches expected for the "Personal Information" dataset, this approach correctly obtained 6 matches only. Likewise, the EMI found 3 out of 4 and 4 out of 6, for the datasets "Inventory" and "Real Estate", respectively.

However, according to the discussion at the end of Section 2.3.1, as well as by observing the column "EMI" marked with "*" in Table 2.5, there are several candidate complex matches that were suggested to the GP phase in each approach. Note that amongst those are the exact remaining matches not found by the EMI technique. This is an indication that, although not sufficient in itself, the EMI approach is an effective pre-processing stage to the GP approach, by reducing the complexity of the search space while providing a high quality list of candidate complex matches.

Column GP of Table 2.4 indicates that, using genetic programming alone, the $F1$ measure obtained was higher, and that all simple mappings were found. However, precision was 0.8 for the "Personal Information" dataset and 0.96 for the "Inventory" dataset, which indicates that some matches were mistakenly suggested.

Table 2.4 shows that our two-phase technique resulted in a considerable improvement over the independent use of the EMI and GP approaches when used independently. This improvement is related to the fact that the first phase, using the EMI matrix, correctly found all simple matches and suggested correct complex matches to the second phase.

The fact that the EMI matrix suggests correlated properties helps reduce the solution space considered by the genetic programming algorithm, thus improving its overall performance. In our tests, the run time of the combined approach showed an improvement of approximately 36% when compared with the run time of the genetic programming approach alone.

As for the second part of the evaluation, we present the results of our method compared against state of the art methods. From previously reported results in terms of accuracy, iMap obtains 0.84 and 0.55 for 1:1 and 1:n mappings respectively, while we obtain 1 and 0.955 for the "Inventory" dataset. For the "Real Estate" dataset, iMap achieves 0.58 and 0.32, whereas we achieve 1 and 0.72, respectively. We also compared our method against LSD [37], which is able to find only simple 1:1 matchings and achieves an accuracy of 0.67 for the "Real Estate" dataset.

2.7 Related Work

In this section, we review the *schema matching* and *ontology matching* literature. Schema and ontology matching rely on the task of automatically finding correspondences between elements or concepts between two or more data models, aiming to create a unified view of data between different sources. Considerable effort of the database community and others have facilitated the integration of heterogeneous

data, (see [38, 39, 40, 41] for traditional surveys). However, as outlined in recent works by Bernstein et al. [42] and Shvaiko and Euzenat [43], there is still much to be done in this field. A list of challenges, future directions and trends are also provided in their work. Accordingly, we focused on finding complex matches, a problem rarely addressed in the literature.

Ontology matching frameworks implement a set of similarity measures to find the correct mappings. For instance, Duan et al. [44] utilize user feedback to determine the importance of each similarity measure in the final mapping result. Similarly, Ritze et al. [45] introduce ECOMatch that uses alignment examples to define parameters to set the correct mapping strategy. Dhamankar et al. [46] describe iMap that predefines modules of functions to semi-automatically find simple and complex matches by leveraging external knowledge. Likewise, Albagli et al. [47] search for mappings using Markov Networks, which combines different sources of evidence (e.g. human experts, existing mappings, etc). Finally, Spohr et al. [48] use a translation mechanism to discover mappings in cross-lingual ontologies. Unlike these works, our approach stands out by creating matching functions automatically. Moreover, most of the approaches depend on a non-trivial manual effort, which we avoid by adopting genetic programming.

A drawback in most approaches is scalability. Duan et al. [49] address the scalability problem using a local sensitivity hashing to match instances inside a cluster. Jiménez-Ruiz and Grau [50] propose an “on the fly” iterative method called LogMap that, based on a set of anchors (exact mappings), creates, extends and verifies mappings using a logical reasoner. Complementary, Wang et al. [51] suggest a method for reducing the number of anchors needed to match ontologies. Likewise, our two-phase approach deals with the problem of scalability by reducing the search space and the need for a low number of instances to find the mapping functions.

Several frameworks have been developed to tackle schema and ontology matching problems. For instance, S-Match [52] is a semantic matching framework for mapping lightweight ontologies. Their approach is based on removing ambiguities introduced by natural language through the use of description logic to relate nodes in different taxonomies. A similar approach is presented by [53]. RiMOM [54] is a framework responsible to find semantic matching between entities in different ontologies using a dynamic strategy to select and combine textual and structural metrics to generate the matching. COMA++ [55, 56] is a multi-strategy and graph-based system able to combine multiple matching algorithms, reuse previous match mappings and support matching between different schemas and ontologies. A new version of this system is under development [57], COMA 3.0, and is expected to support complex matches, which is addressed by our approach [4, 5]. Several

other systems, such as DSSim [58], Anchor-Flood [59], Agreement-Maker [60] and SAMBO [61] tackle the alignment for ontologies and schemas relying on lexical, structural and semantical similarity measures.

Contrasting with the approaches just outlined, we provide an automatic technique that finds simple and complex mappings between RDF datatype properties without prior knowledge that can evolve to adapt to schema and ontology changes.

As for the most related work to our approach, Carvalho et al.[62] propose a genetic programming approach for deduplication problem. However, as the results show, our two-phase approach achieves better results than those using only the genetic programming approach. Moreover, we extend his work to match simple and complex numeric datatype properties. Another similar approach is proposed by Leme et al. [30, 31], where they use apply a similarity-based matching model that uses the Estimated Mutual Information matrix to find simple matches. We adapt and extend their approach to reduce the search space find complex matches between sets of datatype properties.

2.8 Conclusion

In this chapter, we described an instance-based, property matching technique that follows a two-phase strategy. The first phase constructs the estimated mutual information matrix of the property values to identify simple property matches and to suggest complex matches, while the second phase uses a genetic programming approach to detect complex property matches and to generate their property mappings. This combined strategy proved promising to beat combinatorial explosion. In fact, our experiments prove that the technique is a promising approach to construct complex property matches, a problem rarely addressed in the literature.

3

Entity Linking

3.1 Introduction

The emergence of the Linked Data approach has led to the availability of a wide variety of structured datasets on the Web[1] which are exposed according to Linked Data principles [63]. However, while the central goal of the Linked Data effort is to create a well-interlinked graph of Web data, links are still comparatively sparse, often focusing on a few highly referenced datasets such as DBpedia, YAGO [64] and Freebase, while the majority of data exists in a rather isolated fashion. This is of particular concern for datasets which describe the same or potentially *related* resources or real-world *entities*. For instance, within the academic field, a wealth of potentially connected entities are described in bibliographic datasets and domain-specific vocabularies, while no explicit relationships are defined between equivalent, similar or connected resources [65].

Furthermore, knowledge extraction and Named Entity Recognition (NER) tools and environments, such as GATE [66], DBpedia Spotlight[2], Alchemy[3], AIDA[4] or Apache Stanbol[5], are increasingly applied to automatically generate structured data (entities) from unstructured resources such as Web sites, documents or social media. For example, such automatically generated data may provide some initial classification and structure, such as the association of terms with entity types defined in a structured RDF schema (as in [67]). However, entities extracted via Natural Language Processing (NLP) techniques usually are noisy, ambiguous and lack sufficient semantics. Hence, identifying links between related entities within a particular dataset, as well as with pre-existing knowledge, serves three main

[1] http://lod-cloud.net/state
[2] http://spotlight.dbpedia.org/
[3] http://www.alchemyapi.com
[4] http://adaptivedisclosure.org/aida/
[5] http://incubator.apache.org/stanbol

purposes (a) enrichment, (b) disambiguation and (c) data consolidation. Often, dataset providers aim at *enriching* a particular dataset by adding links (*enrichments*) to comprehensive reference datasets. Current interlinking techniques usually resort to mapping entities which refer to the same resource or real-world entity. Recalling the approach presented in Chapter 2, after mapping disparate ontologies, a deduplication[6] process starts to find and create references involving the identical entities between datasets. For instance, `owl:sameAs` references can be created between an extracted entity representing the city "Berlin" with the corresponding Freebase and Geonames[7] entries.

However, additional value lies in the detection of *related* entities within and across datasets. For instance, by creating `skos:related` or `so:related` references between entities that are to some degree connected [2, 3]. In particular, the widespread adoption of reference datasets opens opportunities to analyse such reference graphs to detect the *connectivity*, i.e., the *semantic association* [68, 69] between a given set of entities. However, uncovering these connections would require the assessment of very large data graphs in order to (a) identify the paths between given entities and (b) measure their meaning with respect to a definition of semantic connectivity.

Thus, in this chapter, we present a general-purpose approach that combines a co-occurrence-based and a semantic measure to uncover relationships between entities within reference datasets in disparate datasets. Our novel semantic connectivity score is based on the Katz index [70], a score for measuring relatedness of actors in a social network, which has been adopted and expanded to take into account the semantics of data graphs, while the co-occurrence-based method relies on Web search results retrieved from search engines. Finally, we evaluate the approach using the publicly available USAToday corpus and compare our entity connectivity results with related measures.

The remainder of this chapter is structured as follows. Section 3.2 presents the use case scenario that motivated our approach. Section 3.3 presents our entity connectivity approach. Section 3.4 and Section 3.5 show the evaluation strategies and their results. Section 3.6 discusses previous related work in the field. Finally, Section 3.7 summarises our contributions and discusses the outcomes.

[6] In this context, we refer to the term deduplication as the process of finding duplicates between datasets and not the process of eliminating repeating data.

[7] `http://www.geonames.org`

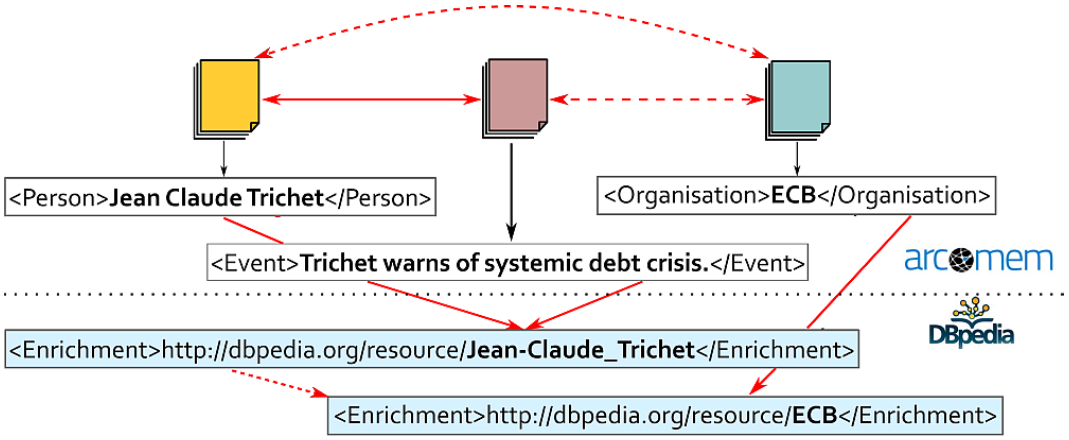


Figure 3.1: Example: connections between Web resources, extracted entities and DBpedia enrichments within ARCOMEM dataset.

3.2 Motivation

We now describe two examples originating from actual Web information integration problems to illustrate the motivation of our work on discovering latent semantic relationships through its semantic relations.

The first example is derived from datasets specific to the ARCOMEM project[8], which primarily consist of extracted information about events and entities (see [71]). ARCOMEM follows a use case-driven approach based on scenarios aimed at creating focused Web archives, particularly of social media, by adopting novel entity extraction and interlinking mechanisms. These archives deploy a document repository of crawled Web content and a structured RDF knowledge base containing metadata about entities and events detected in the archived content.

For instance, Figure 3.1 shows three sets of Web resources (depicted at the top), each associated with one particular entity/event, where the entity ("Jean Claude Trichet") and event ("Trichet warns of systemic debt crisis") are both enriched with the same DBpedia[9] entity (`http://dbpedia.org/resource/Jean-Claude-Trichet`). This allows us to cluster the respective entity and event, and their connected Web resources, as an example of direct connection (solid line in the diagram). However, the third set of Web resources is connected with a third entity ("ECB") which refers to the *European Central Bank*, enriched with the corresponding DBpedia resource (`http://dbpedia.org/resource/ECB`). While NLP and standard IR approaches would fail to detect a connection between them, analysing the DBpedia graph uncovers a close connection between ECB and *Jean Claude Trichet*

[8] `http://www.arcomem.eu`
[9] `http://www.dbpedia.org/`

(being a former ECB president), and hence allows us to establish a connection (dashed line) between all involved entities/events and their connected Web resources.

The second example originates from previous work on integrating biomedical educational Web resources [65]. Here, enrichment was applied to semi-structured metadata of learning resources (composite Web documents) in order to uncover connections amongst educational resources from disparate corpora. Usually, educational resource metadata consists of structured descriptions (XML, RDF) of, for instance, the targeted subjects, skill levels, or learning outcomes. However, free text is still widely used while the adoption of taxonomies (such as MESH[10] or SNOMED[11]) is limited and fragmented. Therefore, this work was dedicated to enriching existing educational metadata with references to cross-domain datasets (Freebase, DBpedia) and domain-specific vocabularies, such as those provided by the BioPortal[12]. This allowed us to automatically cluster resources from distinct native repositories by linking resources which shared equivalent enrichments (e.g. resources which are enriched with the DBpedia entity for cardiology[13]). However, that did not take advantage of the knowledge about entity connections provided by the underlying reference graphs such as DBpedia. For instance, resources enriched with `http://dbpedia.org/resource/Heart_failure` clearly are connected to cardiological resources. Our work, aimed at measuring the connectivity between given enrichments (or entities), contributes to solving this problem.

3.3 An Approach to Entity Linking

In this section, we introduce two novel measures for entity interlinking, a semantic graph-based connectivity score and one which utilises co-occurrence on the Web. Both detect complementary relationships between entities as results show in Section 3.5.

3.3.1 Semantic Connectivity Score (SCS)

In this section, we define a semantic connectivity score between entities, based on a reference graph that describes entities and their relations. Similar to Damljanovic et al. [72], we distinguish between *hierarchical* and *transversal* relations in a given graph. Typical hierarchical properties in RDF graphs are, for instance, `rdfs:subclassOf`, `dcterms:subject` and `skos:broader`, and usually serve as an indicator for similarity between entities. In contrast, transversal properties do not

[10] `http://www.nlm.nih.gov/mesh/`
[11] `http://www.ihtsdo.org/snomed-ct/`
[12] `http://bioportal.bioontology.org/`
[13] `http://dbpedia.org/resource/Cardiology`

indicate any classification or categorisation of entities, but describe non-hierarchical relations between entities which indicate a form of connectivity independent of their similarity.

To illustrate the semantic connectivity, we refer to the pair of entities "Jean Claude Trichet" and "European Central Bank", which have no equivalence or taxonomic relation, but have a high connectivity according to transversal properties. For example, the "European Central Bank" is linked to the entity "President of the European Central Bank" through the RDF property `http://dbpedia.org/property/leaderTitle` that, for its part, links to "Jean Claude Trichet" through the RDF property `http://dbpedia.org/property/title`.

Now, let R be a reference tripleset and G be the associated undirected graph, in the sense that the nodes of G correspond to the individuals occurring in R and the edges of G correspond to the properties between individuals defined in R. From this point on, we will refer to the individuals occurring in R as *entities*.

We define the *semantic connectivity score* (SCS) between a pair of entities (e_1, e_2) in G as follows:

$$SCS(e_1, e_2) = 1 - \frac{1}{1 + (\sum_{l=1}^{\tau} \beta^l \cdot |paths^{<l>}_{(e_1,e_2)}|)} \tag{3-1}$$

where $|paths^{<l>}_{(e_1,e_2)}|$ is the number of *transversal* paths of length l between entities e_1 and e_2, τ is the maximum length of paths considered (in our case $\tau = 4$, as explained in more details below), and $0 < \beta \leq 1$ is a positive damping factor. The damping factor β^l is responsible for exponentially penalizing longer paths. The smaller this factor, the smaller the contribution of longer paths is to the final score. Obviously, if the damping factor is 1, all paths will have the same weight independent of the length. In previous experiments, we observed that $\beta = 0.5$ achieved better results in terms of precision [6]. Equation 3-1 is normalised to range between $[0, 1)$.

Returning to the example presented in Section 3.2, we compute the semantic connectivity score for the entities *"Jean Claude Trichet"* (JCT) and *"European Central Bank"* (ECB), using DBpedia as the reference tripleset. Omitting the details, let us assume that we obtained 8 paths of length 2, and 14 paths of length 3, resulting in the following score:

$$\begin{aligned} SCS(JCT, ECB) &= 1 - \frac{1}{1 + (0.5^2 \cdot 8 + 0.5^3 \cdot 14)} \\ &= 1 - \frac{1}{1 + (2 + 1.75)} = 0.79 \end{aligned} \tag{3-2}$$

Note that even with a small number (i.e., 8) of short paths (of length 2), the contribution to the overall score (2 in Eq. 3-2) is larger than longer paths (1.75 in Eq. 3-2). Evidently, the score obtained by a longer path can overcome a shorter path depending on the number of paths found and the damping factor assigned.

The semantic connectivity score between entities is a variation of the Katz index [70] introduced to estimate the relatedness of actors in a social network. We introduced a number of derivations to improve its applicability to large graphs and to reflect the added semantics provided by labeled edges in RDF graphs, as opposed to the limited semantics of edges in a social network. A detailed discussion of the advantages and limitations of our approach is provided in Section 3.7.

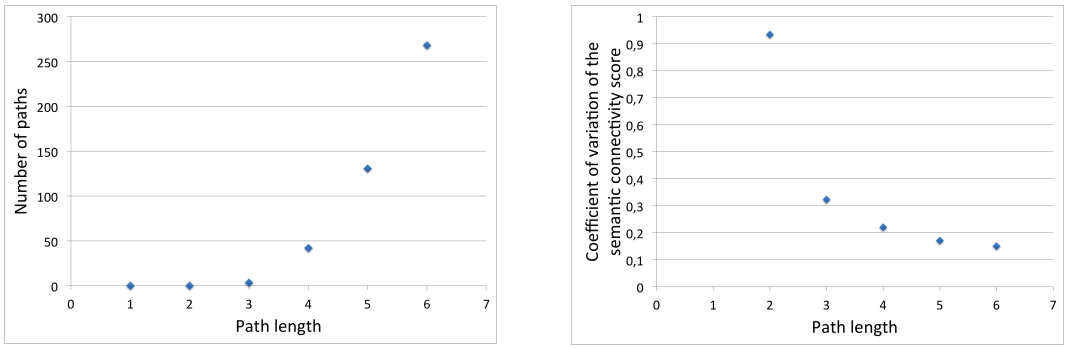
As one main adaptation of Katz, we exploit the semantics of edges in a given data graph by excluding hierarchical properties from our connectivity score computation. As defined earlier, connectivity is indicated by transversal properties. Currently, no further distinction between property types has been introduced into our formula, though we explicitly envisage such an adaptation. However, given the vast amount of property types in datasets such as DBpedia, a distinction at the general and domain-independent level is computationally too expensive and therefore does not scale. Instead, we particularly suggest the adaptation of our formula to specific domains or entity types, which allows the consideration of more fine-grained semantics provided by distinct property types.

In addition, we opted for an undirected graph model in order to reduce computational complexity, since a property is often found in its inverse form (e.g. fatherOf/sonOf) [73]. While most current entity interlinking techniques apply their approaches to a restricted set of entity types to allow some sort of tailoring and, as consequence, more precise results, our experiments in Section 3.5 show that even our fairly generic score produces useful and promising results, which can be improved by means of domain-specific adaptations.

As the semantic score is based on the number of paths and distances (length of a path) between entities, SCS considers only paths with a maximum length ($\tau = 4$), as also adopted in [74]. This maximum length was identified by investigating the semantic score behaviour for edge distances ranging from 1 to 6, as detailed below.

In our experiments, we randomly selected 200 entity pairs and computed the semantic connectivity score (SCS) (see Eq. 3-1) for the aforementioned path length range (see Figure 3.2a). As expected, the average number of paths grows exponentially with the distance (i.e. the path length), see Figure 3.2a.

Thus, as in the small world assumption [75], beyond a certain path length,



(a) (b)

Figure 3.2: Maximum path length analysis. Figure (a) shows the number of paths with respect to length and (b) shows the gain of information when considering different path lengths.

every node pair is likely to be connected. However, as opposed to the small world assumption that people are interlinked through a maximum distance of 6 connections, we found that for interlinking entities this number is lower, approximately by two degrees. This decision is backed up according to several experiments, detailed below.

After computing all entity pairs for different path lengths, we evaluated the coefficient of variation of the semantic score, $C_v = \sigma/\mu$, where, for a given length, σ is the standard deviation of the number of paths and μ is the mean number of paths. This coefficient is used to measure the spread of the semantic score distribution, taking into account an upper bound path length (see Figure 3.2b).

From the behaviour of the curve in Figure 3.2b, it is apparent that the contribution of paths with distances greater than 4 edges is low. Also as expected, the average running time to compute the path grows exponentially with the distance. Hence, including longer path lengths increases significantly the computational costs, while producing only minimal gains in performance. Thus, we obtain the best balance between performance and informational gain to the semantic score. That is, we minimise the path length considered, while maximise the contribution in the overall score.

3.3.2 Co-occurrence-Based Measure (CBM)

We introduce in this section a co-occurrence-based measure between entities that relies on an approximation of the number of existing Web pages that contain their labels. For example, we estimate the CBM score of a pair of entities by submitting queries (such as “Jean Claude Trichet” + “European Central Bank”) to a search engine and retrieving the total number of search results that contain the entity labels in their text body.

Thus, we define the CBM score of a pair of entities e_1 and e_2 as follows:

$$CBM(e_1, e_2) = \begin{cases} 0, \text{ if } count(e_1) = 0 \text{ or } count(e_2) = 0 \text{ or } count(e_1, e_2) = 0 \\ 1, \text{ if } count(e_1) = 1 \text{ or } count(e_2) = 1 \text{ or } count(e_1, e_2) = 1 \\ \frac{Log(count(e_1, e_2))}{Log(count(e_1))} \cdot \frac{Log(count(e_1, e_2))}{Log(count(e_2))}, \text{ otherwise} \end{cases} \quad (3\text{-}3)$$

where $count(e_i)$ is the number of Web pages that contain an occurrence of the label of entity e_i, and $count(e_1, e_2)$ is the number of Web pages that contain occurrences of the labels of both entities. Note that $count(e_1, e_2)$ is a non-negative integer always less than or equal to $count(e_i)$, for $i = 1, 2$. Hence, the final score is already normalised to $0 \leq CBM(e_1, e_2) \leq 1$.

There are other similar approaches to quantify the relation between entities, such as Pointwise Mutual Information (PMI)[76] and Normalised Google Distance (NGD)[77]. However, they take into account the joint distribution and the probability of their individual distributions, which requires to know a priori the total number of Web pages searched by a search engine.

To illustrate the co-occurrence-based score (CBM), consider the values $count(e_1)$ = $count(e_2)$ = $count(e_1, e_2)$, meaning that all occurrences of e_1 and e_2 appear together. In this case, the resulting co-occurrence-based score is 1, disregarding the number of search results.

For example, having $count(e_1)$ = $count(e_2)$ = $count(e_1, e_2)$ = 10 or $count(e_3)$ = $count(e_4)$ = $count(e_3, e_4)$ = 1000, would result in the same score. Evidently, if we would consider the probabilities, as in PMI or NGD, the latter case would get a higher score. Nevertheless, since we are not interested in disjoint comparisons, e.g., $CBM(e_1, e_2)$ against $CBM(e_3, e_4)$, we do not need to estimate the total number of pages, neither include it in the formula.

3.3.3 Towards a Combined Measure

Although there is an overlap between the semantic and co-occurrence based approaches, some relationships cannot be uncovered by co-occurrence methods or by semantic methods alone (see Section 3.5.2). Thus, given that the results from SCS and CBM are seen as complementary, one conclusion is to combine them, which provides the advantage of scalability at discovering entity connections, where CBM would be used as a default approach, and SCS could be employed as an extensive search for finding latent connections in the resulting set of entity pairs

deemed unconnected according to CBM, see Eq. 3-4.

$$\alpha_{CBM+SCS}(e_i, e_j) = \begin{cases} CBM(e_i, e_j), & \text{if } CBM(e_i, e_j) > 0 \\ SCS(e_i, e_j), & \text{otherwise} \end{cases} \quad (3\text{-}4)$$

where e_i and e_j are entities and $i \neq j$.

3.4 Evaluation Setup

3.4.1 Dataset

The dataset for assessing entity connectivity consists of a set of 40,000 document pairs randomly selected from the USAToday news Website[14], where each document contains a title and a summary as textual content. The summary of each document has on average 200 characters. The corpus was annotated using DBpedia Spotlight which resulted in approximately 80,000 entity pairs.

3.4.2 Gold Standard

Given the lack of benchmarks for validating latent relationships between entities, we created a gold standard using CrowdFlower[15], a crowdsourcing platform. To ensure a sufficient quality of the results, we required each user to pass through a set of tests where correct answers were known already, what allowed us to filter out poor assessors. In this way, we were able to avoid relevance judgements from untrusted workers. Moreover, as our corpus is focused on American news, we restrict the assessment only to workers located in the United States.

Thus, in order to construct the gold standard, we randomly selected 1000 entity pairs and 600 document pairs to be evaluated. The evaluation process consisted of a questionnaire in a 5-point Likert scale model where participants are asked to rate their agreement of the suggested semantic connection between a given entity pair. Additionally, we inspected participants' expectations regarding declared connected entities. In this case, presenting two entities deemed to be connected, we asked participants if such connections were expected (from *extremely unexpected* to *extremely expected* in the Likert scale).

The collected judgements provided a gold standard for the analysis of our techniques. Note that in the case of this work, additional challenges are posed with respect to the gold standard, because our semantic connectivity score is aimed at

[14] http://www.usatoday.com
[15] https://www.crowdflower.com/

detecting possibly unexpected relationships which are not always obvious to the user. To this end, a gold standard created by humans provides an indication of the performance of our approach with respect to precision and recall, but it may lack appreciation of some of our found relationships (see Section 3.5.2 for a detailed discussion).

3.4.3 Evaluation Methods

We also present a comparison of our approach against competing methods which measure connectivity via co-occurrence-based metrics to detect entity connectivity. In this evaluation we compared the performance of CBM against SCS and a third method (Explicit Semantic Analysis (ESA)) that is based on statistical and semantic methods.

Specifically, ESA [78] measures the relatedness between Wikipedia concepts by using a vector space model representation, where each vector entry is assigned using the *tf-idf* weight between the entities and its occurrence in the corresponding Wikipedia article. The final score is given by the cosine similarity between the weighted vectors. Note that ESA can be applied to measure any kind of corpora, not just Wikipedia concepts.

3.4.4 Evaluation Metrics

We measure the performance of the entity connectivity using the standard metrics of precision (P), recall (R) and $F1$ measure. Note that in these metrics, as relevant entity pairs, we consider those that were marked in the gold standard (gs) as connected according to the 5-point Likert Scale (*Strongly Agree* & *Agree*).

(P) is defined as the ratio of the set of retrieved entity pairs that have relevant uncovered connections over the set of entity pairs that have connections, see Eq. (3-5).

$$P = \begin{cases} 0, \text{ iff } |\mu^{\tau}_{retrieved}| = 0 \\ \\ \dfrac{|\mu^{\tau}_{retrieved} \cap \mu_{relevant}|}{|\mu^{\tau}_{retrieved}|}, \text{ otherwise} \end{cases} \tag{3-5}$$

where $\mu_{relevant}$ is the set of retrieved entity pairs that are relevant and $\mu^{\tau}_{retrieved}$ is the set of retrieved connections that has a semantic connectivity score greater than a given threshold (τ). The threshold used in our experiments is shown in Section 3.5.

The recall measure is the ratio of the set of the retrieved entity pairs (R) that have relevant uncovered connections over all relevant connected entity pairs according to the gold standard, see Eq. (3-6).

$$R = \begin{cases} 0, \text{ iff } |\mu_{relevant_{(gs)}}| = 0 \\ \\ \dfrac{|\mu^{\tau}_{retrieved} \cap \mu_{relevant}|}{|\mu_{relevant_{(gs)}}|}, \text{ otherwise} \end{cases} \tag{3-6}$$

where $\mu_{relevant_{(gs)}}$ is the set of all relevant entity pairs.

Finally, $F1$ measure shows the balance between precision and recall, and is computed as:

$$F1 = \begin{cases} 0, \quad \text{iff } |P + R| = 0 \\ \\ 2 \cdot \dfrac{P \cdot R}{P + R}, \text{ otherwise} \end{cases} \tag{3-7}$$

3.5 Results

For each method described in the Sections 3.3 and 3.4, we present the results on their ability to discover latent connections over the entities. Furthermore, we also present an in depth-analysis of their shortcomings and advantages for discovering connections between entities.

3.5.1 Entity Connectivity Results

Table 3.1 shows the results obtained by the questionnaire and used as gold standard for the entity connectivity. The results are presented in a 5-point Likert scale of agreement ranging from *Strongly Agree* to *Strongly disagree*.

Table 3.1: Number of entity-pairs in each category (5-point Likert scale) in gold standard.

Strongly Agree	Agree	Undecided	Disagree	Strongly Disagree
63	178	127	227	217

In Figure 3.3, we report the performance for the co-occurrence-based score (CBM), Explicit Semantic Analysis (ESA) and our proposed adaptation of the Katz score (SCS). We considered as relevant all the entity pairs which had relevance judgements as *Strongly Agree* and *Agree*, and scores greater than a threshold. Since

our task is to uncover latent relationships between entities rather than ranking them, we set the threshold to 0 (i.e. we include all results), but for some tasks we might want to raise this, e.g. for ranking or recommending.

According to Figure 3.3, SCS performs better in terms of precision whereas CBM achieves highest recall value. SCS and CBM present only minimal differences with respect to precision and recall, while ESA has the lowest values for all metrics.

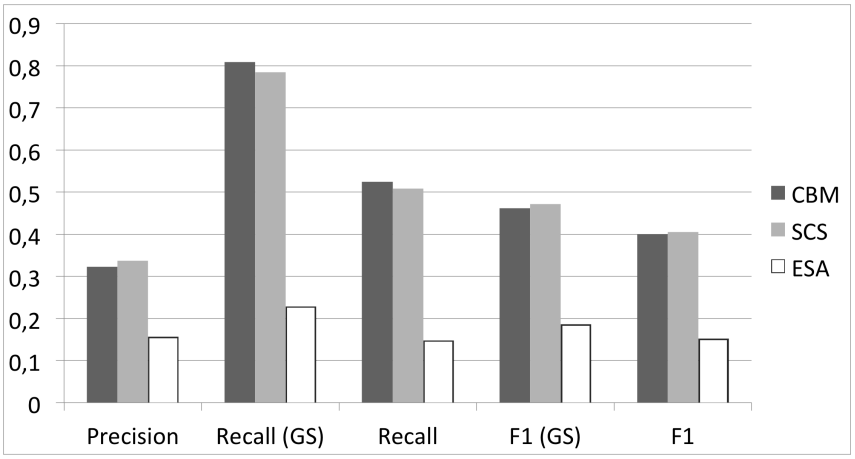


Figure 3.3: P/R/F1 measure according to the gold standard (GS) amongst methods.

In addition to performance, we are also interested in the agreement between the methods. Identifying missed and detected relationships amongst all measures provides an indicator of their complementarity. In Table 3.2 we present a pairwise comparison of methods where we show the ratio of connections that are found by one method and missed by another. It is notable that CBM and SCS capture most of the connections, even though CBM misses 3.1% and 11.2%, and SCS misses 9.5% and 12.3% for *Strongly Agree* and *Agree* respectively.

Table 3.2: Ratio of connections detected by each method, according to the gold standard.

	CBM (not in SCS)	CBM (not in ESA)	SCS (not in CBM)	SCS (not in ESA)	ESA (not in CBM)	ESA (not in SCS)
Strongly Agree	**9.5%**	**76%**	**3.1%**	**71%**	**7.9%**	**9.5%**
Agree	**12.3%**	**63.4%**	**11.2%**	**60.1%**	**8.9%**	**6.7%**
Undecided	9.4%	60.6%	6.3%	59.8%	5.5%	7.9%
Disagree	15.0%	63.0%	7.1%	53.3%	7.1%	5.3%
Strongly Disagree	18.4%	63.1%	51.6%	4.6%	4.6%	6.9%

Besides the missed connections, we also take into account the expectedness of a connection between entity pairs. The expectedness shows how well established the connection is: an unexpected connection would be a relevant inferred indirect

link between the entities. Thus, unexpectedness can be interpreted as a creation of novel links between entities. We see that SCS uncovers 25% of the unexpected connections, while CBM uncovers 16%. For this task, ESA was not able to uncover any new connections.

3.5.2 Results Analysis

In this section, we provide a detailed analysis of the results. The analysis is guided by the initial aims of our work on discovering latent connections between entities within a data graph (at varying path lengths), rather than competing with well established methods such as co-occurrence-based approaches widely deployed by search engines. To this end, the results of the listed approaches are complementary, where each of the approaches is able to establish unique entity connections.

In Figure 3.4, we show the agreement of entity pair ranking retrieved by SCS compared with CBM. The entity pair ranking follows an expected decline, where most connections are found at high ranks, whereas only a few are found at very low ranks.

As we can see in Figure 3.4, for the topmost rank of co-occurrence-based entity pairs, 225 of them have a semantic connection. Ideally, since these pairs are ranked in the top position, we expected to find a semantic connection between all of them. Arguably, the dependency rank-position to semantic connection should follow the trend where the lower the rank position, the higher the number of semantic

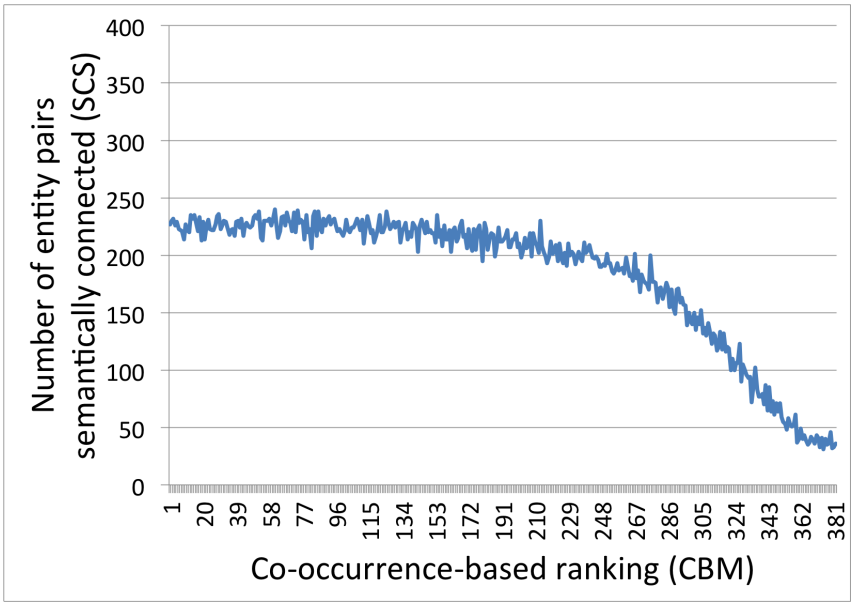


Figure 3.4: The x-axis represents the ranking position *x* of entity pairs according to the CBM rankings. The y-axis represent the number of entity pairs ranked at *xth* position that have a semantic connection according to our connectivity threshold.

Table 3.3: Kendall tau and Jaccard-index between SCS and CBM entity rankings.

	k@2		**k@5**		**k@10**	
Dataset	Kendall tau	Jaccard-index	Kendall tau	Jaccard-index	Kendall tau	Jaccard-index
USAToday	0.40	0.09	0.47	0.19	0.52	0.21

connected entity pairs. In this sense, we can estimate which items have some missing relations. This is the first step in the task of actually discovering the missing relations. By observing the missing semantic ranked pairs on the x-axis, we can identify which entities miss some connection induced by the co-occurrence-based score (the problem introduced on Section 3.2). It is worth noting that, after the 260th rank position in the x-axis, the behaviour of the curve is in line with our expectations, i.e., the lower the correlation induced by the co-occurrence-based score, the lower that induced by the semantic connectivity score.

To show the complementarity between CBM and SCS, we used the Kendall tau rank correlation coefficient to assess the agreement of the entity ranks induced by the semantic connectivity score based on the DBpedia graph against the entity ranks induced by CBM. Table 3.3 shows the results.

As we can see from Table 3.3, the overlap between the rankings is not high. However, as our previous evaluation with the gold standard shows, this indicates that the scores induce different relationships between entities. The CBM score induces a relationship that reflects the overall co-occurrence of entities in the Web, whereas the semantic connectivity score mirrors the DBpedia graph.

Thus, as shown in Table 3.4, the CBM+SCS is the best performing approach compared to the other methods for the task of entity connectivity. Moreover, when comparing the $F1$ results from the CBM+SCS and SCS, we achieve significantly different results for p-value $= 0.04$ with 95% confidence.

Table 3.4: P/R/F1 measures according to gold-standard and amongst methods.

	CBM	SCS	ESA	CBM+SCS
Precision	0.32	0.34	0.16	0.34
Recall (GS)	0.81	0.78	0.23	0.90
Recall	0.52	0.51	0.15	0.58
F1 (GS)	0.46	0.47	0.19	0.50
F1	0.40	0.41	0.15	0.43

We would also like to point out the challenges posed by our approach on creating a gold standard. As mentioned previously, while our work aims at detecting semantic entity connections beyond traditional co-occurrences, this results in connections which might be to some extent unexpected yet correct, according

to background knowledge (such as DBpedia in our case). Hence, using a manually created gold standard, though being the only viable option, necessarily impacts the precision values for our work in a negative way, as correct connections might have been missed by the evaluators. This has been partially confirmed by the large number of detected co-occurrences which were marked as *undecided* by the users, where manual inspection of samples in fact confirmed a positive connection. This confirms that in a number of cases, connections were not necessarily incorrect but simply unknown to the users. Thus, we believe that a more thorough evaluation providing the evaluators with information on how a connection emerged, by showing all properties and entities that are part of a path greater than one, would give us more reliable judgements.

An example found in our evaluation is between the politicians “Barack Obama” and “Olympia Snowe”, where the first is the current US president and the latter is one of the current senior US senators. Although the evaluators did not identify a connection between them, our semantic connectivity approach found several paths with length 2 or more. Additionally, they are related via several topics in real life, which confirms the validity of the paths found by our approach. For instance, this information could be exploited by news Websites for improving the user experience on finding related topics or news.

3.6 Related Work

Lehmann et al. [79] introduce RelFinder, which shows semantic associations between multiple entities from a RDF dataset, based on a breadth-first search algorithm, that is responsible for finding all related entities in the tripleset. Contrasting with RelFinder, Seo et al. [80] proposed the OntoRelFinder that uses a RDF Schema for finding semantic associations between two entities through its class links. Scarlet [81, 82] is another approach that relies on different schemas to identify relationships between entities.

Han et al. [83] propose a slightly different approach. Instead of finding connections between two given entities, they expect to find the entities that are most connected, with respect to a given relationship and entity. This approach is interesting since it throws another perspective on the problem that we consider. However, they look for connected entities by means of a known relationship, while we aspire to uncover such connections between known entities.

Anyanwu et al. [84] present the SemRank, a customizable query framework that allows different setups for ranking methods, resulting in different perspectives for the same query. Thus, given two entities, depending on the setup the search

results vary from more traditional (e.g. common connections or closest paths between entities) to less traditional (e.g. longer paths). In our approach, we consider both short and long paths to determine connectivity between two entities and Web resources.

Work from Leskovec et al. [85] present a technique suggesting positive and negative relationships between people in a social network. This notion is also addressed in our method, but we take into account the path length. The longer is the path, the smaller is its contribution to the score.

The problem of discovering relationships between entities was also addressed by Damljanovic et al. [72] in Open Innovation scenarios, where companies outsource tasks on a network of collaborators. Their approach exploits the links between entities extracted from both the user profiles and the task descriptions in order to match experts and tasks. For this task, they use reference datasets and distinguish between entities as hierarchical and transversal. Following their approach, we distinguish between both relations types, although we focus on transversal relations.

In a similar vein, Gionis et al. [86] present a framework that basically estimates entity relevance by a set-cover formulation along with entity ranking and entity selection methods. Although they do not take into account the links between entities, they compute the importance of an entity by counting its occurrences in different sets. Fang et al. [74] introduces the REX system, which computes a ranked list of entity pairs to describe entity relationships. The graph structure is decomposed for an entity pair resulting in unique graph patterns and ranks, where these patterns are matched according to a measure of interestingness, based on the traditional random walk algorithm and the patterns found between an entity pair. Sieminski [87] presents a method to measure the semantic similarity between texts on the Web, which consists of a modified *tf-idf* model and semantic analysis that makes use of WordNet structure. However, unlike his work, we explore the connections given by transversal properties in order to uncover latent connections between texts, rather than to explore similarity between them.

3.7 Conclusion

We have presented a general-purpose approach to discover relationships between entities, utilising structured background knowledge from reference graphs as well as co-occurrence of entities on the Web. To compute entity connectivity, we first introduced a semantic-based entity connectivity approach (SCS), which adapts a measure from social network theory (Katz) to data graphs, in particular Linked Data. We were able to uncover 14.3% entity connections not found by the state of the art

method described here as CBM. While using a combination of CBM+SCS, we achieved a $F1$ measure of 43% for entity connectivity.

Our experiments show that SCS enables the detection of entity relationships that a priori linguistic and co-occurrence approaches would not reveal. Contrary to the latter, SCS relies on semantic relations between entities as represented in structured background knowledge, captured in reference datasets.

While both approaches (CBM and SCS) produce fairly good indicators for entity and document connectivity, an evaluation based on Kendall's tau rank correlation showed that the approaches differ in the relationships they uncover [6]. A comparison of agreement and disagreement between different methods revealed that both approaches are complementary and produce particularly good results in combination with each other. The semantic approach is able to find connections between entities that do not necessarily co-occur in documents (found on the Web), while the CBM tends to emphasise entity connections between entities that are not necessarily strongly connected in reference datasets. Thus, a combination of our semantic approach and traditional co-occurrence-based measures provide promising results for detecting related entities.

Despite the encouraging results, one of the key limitations of our Katz-based measure is the limited consideration of edge semantics in its current form. At the moment, property types are distinguished only at a very abstract level, while valuable semantics about the meaning of each edge (i.e., each property) is left unconsidered during the connectivity computation. We are currently investigating approaches to take better advantage of the semantics of properties in data graphs.

Another issue faced during the experimental work is related to the high computational demands when applying our approach to large-scale data, which restricted our experiments to a limited dataset. In particular, the combination of traditional measures with our approach could help in improving performance, for instance, by computing our semantic connectivity only between entity pairs deemed unconnected by traditional measures. In addition, reducing the gathering of paths to a limited set of nodes ("hub nodes") might help in further improving scalability.

4

Document Linking

4.1 Introduction

User-generated content is characterized by a high degree of diversity and heavily varying quality. Given the ever increasing pace at which this form of Web content is evolving, adequate preservation and detection of correlations has become a cultural necessity. Extraction of entities from Web content, in particular social media, is a crucial challenge in order to enable the interlinking of related Web content, semantic search and navigation within Web archives, and to assess the relevance of a given set of Web objects for a particular query or crawl [17].

Traditional approaches to finding related documents are often addressed using a combination of Information Retrieval (IR) and Natural Language Processing (NLP) techniques. These techniques compute the similarities between a set of terms from specific resources based on their overlap, or through latent semantic analysis [88] measuring relatedness of individual terms and documents. Nonetheless, most of these techniques require large corpora and a partially common vocabulary/terminology between the resources. Thus, in such cases, they fail to detect latent semantic relationships between documents.

On the other hand, semantic approaches exploit knowledge defined in a data graph to compute notions of similarity and connectivity. Again, our approach explicitly targets *connectivity* as a measure of the *relationship* between two documents, as opposed to their *similarity*.

In this chapter, we expand the relationship assessment methodology, presented earlier in Chapter 3, to measure the connectivity between documents and hence identify connected and related Web resources. As the results will show, our approach has the ability to expose relations that traditional text-based approaches fail to

identify. We validate and assess our proposed approaches through an evaluation on a real-world dataset, where results show that the proposed technique outperform state of the art approaches.

The remainder of this chapter is structured as follows. Section 4.2 introduces a real-world motivation example that inspired our approach. Section 4.3 introduces the semantic connectivity score for documents and a processing chain to uncover semantic connections between documents. Sections 4.4 and 4.5 show the evaluation method and the outcomes of our method along with an in-depth analysis of the results. Section 4.6 summarises related literature. Finally, Section 4.7 presents the final remarks and advances of our work.

4.2 Motivation

In this section we describe an example to illustrate the motivation behind our work on uncovering *latent semantic relationships* between documents through its *semantic relations*.

The example below shows two descriptions of documents extracted from the USAToday corpus. Note that, the underlined terms refer to the recognised entities in each document derived from an entity recognition and enrichment process.

(i) The Charlotte Bobcats could go from the NBA's worst team to its best bargain.

(ii) The New York Knicks got the big-game performances they desperately needed from Carmelo Anthony and Amar'e Stoudemire to beat the Miami Heat.

Although both documents are clearly related to Basketball/Sports topics, linguistic and statistical approach would struggle to point out that both documents are connected. First, both textual descriptions are rather short and lack sufficient contextual information what makes it harder for purely linguistic or statistical approaches to detect their connectivity. Second, in this particular case, there are no significant common words between the documents.

Usually, statistical and linguistic approaches are particularly suitable for cases where large amounts of textual content is available to detect the relationships between Web resources. In particular, some common terminology is required for detecting similarities between Web resources.

Conversely, these challenges can be partially overcome by taking advantage of structured background knowledge to disambiguate and enrich the unstructured textual

information. The example shows two documents, each associated with a particular entity, where the term *Charlotte Bobcats* was enriched with the entity `http://dbpedia.org/resource/Charlotte_Bobcats` in the document (i) and the term *Carmelo Anthony* was enriched with the entity `http://dbpedia.org/resource/Carmelo_Anthony` in the document (ii).

Thus, analysing the DBpedia graph uncovers a connection between *Charlotte Bobcats* and *Carmelo Anthony* (being a basketball team and player, respectively) and hence allows us to establish a connection between the entities and their connected Web resources. Specifically, both entities are connected through the following path:

Charlotte Bobcats $\leftrightarrow$ **NBA** $\leftrightarrow$ **New York Knicks** $\leftrightarrow$ **Carmelo Anthony**

where the intermediary entities uncover a connection between *Charlotte Bobcats* and *Carmelo Anthony*.

4.3 An Approach to Document Linking

With the help of the previous approaches, we now present the main steps of a processing chain that allows us to identify latent connections between documents in disparate datasets and document corpora. The whole process is composed of the following steps:

S1. *Entity Extraction* – pre-processing of documents for finding and extracting term references and named entities;

S2. *Entity Enrichment* – matching of references in external knowledge bases such as DBpedia and Freebase;

S3. *Entity Connectivity* – uncovering of latent relationships between entities and induction of connections amongst entities;

S4. *Document Connectivity* – uncovering latent relationships between documents through entity connections and inducing connections amongst documents.

Basically, steps S1 and S2 are responsible for extracting structured information from documents (unstructured data) and link to external knowledge bases, while step S3 is dedicated to uncover the relationships between the extracted entities.

For the steps S1 and S2, several tools, such as WikipediaMiner [89] and DBpedia Spotlight [90], have been widely used in academic and industry systems. Particularly, we used DBpedia Spotlight to cover these steps, since it is more suitable for general-purpose approaches like ours. As for the step S3, we applied the approach

previously introduced in Chapter 3. Thus, in this section, we focus on step S4, in which we uncover latent connections between documents.

4.3.1 Semantic Connectivity Score for Documents

In this section, we define a semantic connectivity score for documents which relies on connections between entities based on reference graphs. Based on the semantic connectivity score (SCS) between entity pairs (see Eq. 3-1), we then define the *semantic connectivity score* (SCS_d) between two documents D_1 and D_2 as follows:

$$SCS_d(D_1, D_2) = \begin{cases} 0, & \text{iff } |E_1| = 0 \text{ or } |E_2| = 0 \\ \left(\sum_{\substack{e_1 \in E_1 \\ e_2 \in E_2 \\ e_1 \neq e_2}} SCS(e_1, e_2) + |E_1 \cap E_2| \right) \cdot \dfrac{1}{|E_1| \cdot |E_2|}, & \text{otherwise} \end{cases} \tag{4-1}$$

where E_i is the set of entities found in D_i, for $i = 1, 2$. Note that the score is normalised between [0,1]. The score $SCS_d(D_1, D_2)$ is 0 when no common connection between entity pairs across documents exists, $|E_1| = 0$ or $|E_2| = 0$. Otherwise, the score is represented by the sum of semantic connectivity scores between entities, normalised over the total number of entity pair comparisons.

To illustrate the semantic connectivity score between document pairs, we recall to the motivation example presented in Section 4.2, where the following two descriptions of documents were extracted from the USAToday corpus.

(i) The Charlotte Bobcats could go from the NBA's worst team to its best bargain.

(ii) The New York Knicks got the big-game performances they desperately needed from Carmelo Anthony and Amar'e Stoudemire to beat the Miami Heat.

We observe that the underlined terms are entities previously recognised through the entity recognition and enrichment process (steps S1 and S2).

Thus, for each entity in document (i) and document (ii), we first compute the semantic connectivity score (SCS) between entities. Table 4.1 summarises the scores between entity pairs across documents (i) and (ii).

Thus, the final score between the documents (i) and (ii) is:

Table 4.1: Semantic connectivity scores between entity pairs in document (i) and (ii).

Entities from document (i)	Entities from document (ii)	SCS
Charlotte Bobcats	*New York Knicks*	0.87
Charlotte Bobcats	*Carmelo Anthony*	0.63
Charlotte Bobcats	*Amar'e Stoudemire*	0.60
Charlotte Bobcats	*Miami Heat*	0.89
NBA	*New York Knicks*	0.85
NBA	*Carmelo Anthony*	0.60
NBA	*Amar'e Stoudemire*	0.63
NBA	*Miami Heat*	0.87

$$\begin{aligned} SCS_d(D_1, D_2) &= \frac{(0.87 + 0.63 + 0.60 + 0.89)}{2 \cdot 4} + \\ &+ \frac{(0.85 + 0.60 + 0.63 + 0.87)}{2 \cdot 4} \\ &= \frac{5.96}{8} = 0.74 \end{aligned} \tag{4-2}$$

4.4 Evaluation Setup

In this section, we describe in detail the evaluation methodology and experiment setup used to validate our hypothesis of uncovering latent relationships between documents using the semantic connectivity score SCS_d.

4.4.1 Dataset

As in Section 3.4.1, we used a subset of randomly selected documents extracted from the USAToday news Website to evaluate our approach. In total, we observed 40,000 document pairs, where each document has 200 characters long on average.

Note that our dataset consists of short-length documents, since we intend to reveal latent connections between documents and, as shown further, non-semantic approaches are more likely to fail in short-length documents. Hence, we can clearly demonstrate the benefits of semantic approach.

4.4.2 Gold Standard

In order to validate the results of our evaluation, the first step is to obtain a gold standard of relationships between documents. As in Section 3.4.2, the user evaluation

was set up in CrowdFlower[1], where we conducted a user evaluation to collect user judgements with the aim of creating the gold standard.

Thus, we randomly selected 600 document pairs from the dataset that will be part of the gold standard. The evaluation process consisted of a questionnaire on a 5-point Likert scale model where participants were asked to rate their agreement of the suggested semantic connection between a given document pair.

Furthermore, we inspected participants' expectations regarding declared connected document. We asked participants if such connections were expected (from *extremely unexpected* to *extremely expected*, also on a 5-point Likert scale).

4.4.3 Evaluation Methods

To emphasise the benefits of measuring connectivity between documents using our approach, we compared it against competing methods which measure connectivity via co-occurrence-based metrics to detect entity and document connectivity. In the first evaluation, we compared the performance of SCS_d against two methods: Co-occurrence-based method (CBM) and Explicit Semantic Analysis (ESA) (see Section 3.4.3).

We recall that the co-occurrence-based method (CBM) is a co-occurrence-based score between entities that relies on an approximation of the number of existing Web pages that contain these entities. As we use entities to measure the connectivity between documents, we adapted CBM to work with document pairs instead of entity pairs. Thus, by a small adjustment in Eq. (4-1), we transform the SCS_d function into CBM_d as follows:

$$CBM_d(D_1, D_2) = \begin{cases} 0, & \text{iff } |E_1| = 0 \text{ or } |E_2| = 0 \\ \left(\sum\limits_{\substack{e_1 \in E_1 \\ e_2 \in E_2 \\ e_1 \neq e_2}} CBM(e_1, e_2) + |E_1 \cap E_2| \right) \cdot \dfrac{1}{|E_1| \cdot |E_2|}, & \text{otherwise} \end{cases} \tag{4-3}$$

where E_i is the set of entities found in D_i, for $i = 1, 2$ and the final score is normalised between [0,1].

In addition to ESA and CBM, we also evaluate the document connectivity with the traditional statistical *tf-idf* method, which measures the importance of a

[1] https://www.crowdflower.com/

term to a document in a document corpora. To measure the connectivity between documents, each document is represented by a weighted term vector model computed using *tf-idf*. Finally, the similarity between the (documents) vectors is given by the cosine metric (see Eq. 2-2).

4.4.4 Evaluation Metrics

For measuring the performance of the document connectivity approaches, we used standard evaluation metrics like precision (P_d), recall (R_d) and $F1_d$ measure. Note that in these metrics, as relevant pairs, we consider those marked in the gold standard (gs) as connected according to the 5-point Likert Scale (*Strongly Agree & Agree*).

For the document connectivity, the precision measure (P_d) is the ratio of the set of all retrieved document pairs deemed as relevant over the set of document pairs that are connected. Thus, the relevant documents are those that were marked as *Strongly Agree & Agree*, while the set of document pairs that are connected consists of those that have a semantic connectivity score greater than a given threshold (see Equation (4-4)).

$$P_d = \begin{cases} 0, & \text{iff } |\Phi^{\tau}_{retrieved}| = 0 \\ \\ \dfrac{|\Phi^{\tau}_{retrieved} \cap \Phi_{relevant}|}{|\Phi^{\tau}_{retrieved}|}, & \text{otherwise} \end{cases} \tag{4-4}$$

where $\Phi_{relevant}$ is the set of retrieved document pairs that are relevant and $\Phi^{\tau}_{retrieved}$ is the set of all connected document pairs greater than a given threshold (τ).

The recall (R_d) is the ratio of the set of retrieved documents that are relevant over the set of all relevant document pairs according to the gold standard (see Equation (4-5)).

$$R_d = \begin{cases} 0, & \text{iff } |\Phi_{relevant_{(gs)}}| = 0 \\ \\ \dfrac{|\Phi^{\tau}_{retrieved} \cap \Phi_{relevant}|}{|\Phi_{relevant_{(gs)}}|}, & \text{otherwise} \end{cases} \tag{4-5}$$

where $\Phi_{relevant_{(gs)}}$ is the set of all relevant document pairs.

Finally, $F1_d$ measure shows the balance between precision and recall, and is computed as in Eq. (4-6).

$$F1_d = \begin{cases} 0, \text{ iff } (P_d + R_d) = 0 \\ 2 \cdot \dfrac{P_d \cdot R_d}{P_d + R_d}, \text{ otherwise} \end{cases} \quad (4\text{-}6)$$

4.5 Results

In this section, we report evaluation results for the document connectivity approaches. For each method, we present the results for their ability to discover latent connections between pairs of resources. Furthermore, we also present an in-depth analysis of their shortcomings and advantages for discovering connections between documents.

4.5.1 Document Connectivity Results

Table 4.2 shows the results according to the gold standard presented in the Likert scale, where users evaluated if a given entity pair could be connected in a document. Compared with the gold standard, 368 entity pairs out of 812 could have some connection.

From the set of entities that could co-occur in a document, 51% of those entities were also connected based on our gold standard, while 34% were *Undecided*. Analysis of the results for the *Undecided* category will be provided in Section 4.5.2, since these results are of particular interest in establishing latent relationships between documents.

Table 4.2: Total number of results for the GS in Likert-scale.

Strongly Agree	Agree	Undecided	Disagree	Strongly Disagree
96	272	139	165	140

The performance of each method is shown in Figure 4.1. As in the task of entity connectivity, SCS_d performs slightly better than CBM_d in terms of precision, while CBM_d is better in terms of recall. $F1_d$ measure is similar, with 60.0% and 59.6% for SCS_d and CBM_d, respectively. In both cases, ESA has the lowest performance.

The positive correlation of entity connectedness and their co-occurrence in the same document was 79.6%, 78.0% and 23.5% for SCS_d, CBM_d and ESA respectively, considering only the *Strongly Agree* and *Agree* relevance judgement results.

As already indicated in the introduction in Section 4, our proposed semantic approach can be exploited to measure document connectivity by taking into account

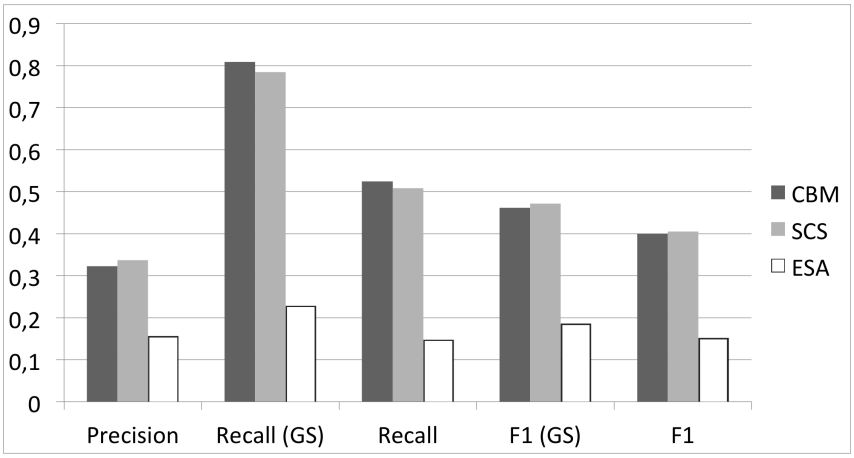


Figure 4.1: P/R/F1 measure according to the gold standard (GS) amongst methods.

the connectedness of entities that describe a document and their semantic connections. Indeed, as shown by the positive correlation of entity connectivity and entity co-occurrence in a document, we claim that our approach can be used as method for inferring document "relatedness" where other statistical models would fail.

To validate the usefulness of our approach, we compared the results against the well established document relatedness measure *tf-idf*. Our approach was able to find 500 unique connections between documents, whereas *tf-idf* found only 25. As described in Section 4.4.1, our corpus is composed of small descriptions of the news articles, which severely limits the ability of *tf-idf* to identify connections between them.

We also conducted an experiment to evaluate the uncovered connections by both methods. We found that 16% of the connections found by our approach were relevant, compared with 12% using *tf-idf*. We took into consideration that the recall achieved by *tf-idf* is only 3.6%, whereas for SCS_d, it is close to 86%.

4.5.2 Analysis of the Results

Table 4.3 shows the results for the task of document connectivity. The mixed approach CBM_d+SCS_d performs best on finding the co-occurrence of entity pairs in a document. It is worth noting as well that the co-occurrence of entity pairs for documents can be retrieved with high recall (89%) when using the proposed combination of CBM_d+SCS_d.

A positive correlation of entity connectivity and co-occurrence in a document is of high importance for our proposed approach, allowing to establish newly constructed knowledge that can be represented as an aggregate of the entity connections.

Table 4.3: Precision, recall and F1 measure amongst methods.

	CBM_d	SCS_d	ESA	CBM_d+SCS_d
Precision	0.47	0.49	0.21	0.51
Recall (GS)	0.80	0.77	0.25	0.89
Recall	0.49	0.48	0.15	0.54
$F1_d$ (GS)	0.59	0.60	0.23	0.64
$F1_d$	0.48	0.48	0.18	0.52

As previously discussed in Section 3.5.2, the creation of the gold standard for this task also depends on the background knowledge of the users. Although some connections between documents are simple to identify, as the one presented in the motivation example (see Section 4.2), some others are not for all users, which hinders the evaluation process. To exemplify, after a manual inspection of document pairs judged as *undecided*, many documents were actually considered connected. Thus, we believe that if we provide additional information, such as document categories, it would facilitate the evaluation process. However, additional information would also lead to a biased evaluation, which we avoided with the current evaluation.

4.6 Related Work

Related work in the field of recommender systems includes the work by Passant [91], which presents a linked data semantic distance measure (LDSD) for music recommendation, by taking mainly into account incoming and outgoing links as well as indirect links between resources (i.e., songs and singers) to determine a recommendation score, used for recommending both direct and lateral music. In later work [92], he introduces a filtering step, by removing properties between resources that are not meaningful in the music context. Work on movie recommendation by Souvik et al. [93] considers an approach based on object features in order to improve movie recommendation, by using several similarity functions that deal with nominal, boolean and numeric features. Furthermore, they also use a linear regression method to assign weights for each feature type. Although this method presents good results, they do not consider semantic connections to uncover latent features.

Kaldoudi et al. [94] discuss how to apply the overall approach of actor/network theory to data graphs. Graph summarization is an interesting approach to exploit semantic knowledge in annotated graphs. Thor et al. [95] exploited this technique for link prediction between genes in the area of Life Sciences. Their approach relies on the fact that summarisation techniques can create compact representations of the original graph, by adopting a set of criteria for creation, correction and deletion of

edges and grouping of nodes. Thus, a prediction function ranks the edges with the most potential, and then suggests possible links between two given genes.

Potamias et al. [96] present another approach based on Dijkstra's shortest path along with random walks in probabilistic graphs to define distance functions that identify the k closest nodes from a given source node.

In the field of Social Networks, Hasan and Zake [97] present a survey of link prediction techniques, where they classify the approaches into the following categories: *feature based link prediction*, *bayesian probabilistic models*, *probabilistic relational models* and *linear algebraic methods*. According to this classification, our approach can be classified as a *feature based link prediction* method. Work from Leskovec et al. [85] presents a technique suggesting positive and negative relationships between people in a social network. This notion is also addressed in our method, but we take into account the path length.

Finding semantic relationships between two given entities is also discussed in the context of ontology matching [98, 99, 100]. In our case, hub ontologies could also be used to infer missing relationships into another ontology.

From the approaches outlined, we combine different techniques to uncover connections between disparate entities, which allows us to exploit the relationships between entities to further identify connected documents.

4.7 Conclusion

In this chapter, we have presented a processing chain to discover document connectivity. Moreover, we have adapted and extended the semantic-based entity connectivity approach (SCS) to interlink documents (SCS_d). SCS_d was able to uncover 16% of unique inferred document connections based on entity co-occurrence, not found by the state of the art method CBM_d. Additionally, while using a combination of CBM_d+SCS_d we achieved an $F1_d$ measure of 52%.

Based on experiments, we verify that the proposed approach (SCS_d) is able to uncover connections that linguistic and co-occurrence approaches are unable to detect. Moreover, the use of reference knowledge bases, such as DBpedia, showed to be crucial in the document linkage process, since the exploration of such knowledge bases reveals information that support our approach.

We also would like to outline that the combination of our semantic approach and traditional co-occurrence-based measures provided very promising results for detecting connected documents as shown by the combination of CBM_d and SCS_d.

5

An Application of Document Linking

5.1 Introduction

The huge amount of Web data and resources, particularly in the academic area, calls for strategies to analyse and explore resources and data.

While scientific disciplines are very data- and knowledge-intensive, the lack of semantic tools hampers information management and decision making. This includes scientific data as well as unstructured academic publications as one of the key outcome of scientific work. This is due to information access offered by digital library providers such as ACM Digital Library[1] and Elsevier[2] being mostly based on free text search and hierarchical classification[3].

In this chapter, as a result of the methods previously presented, we introduce a novel Web application called *Cite4Me* that leverages Semantic Web technologies to provide a new perspective on search and retrieval of Linked Open Data sets (LOD). *Cite4Me* is implemented over a specific bibliographical dataset provided by the *LAK Challenge 2013*[4], where our application was awarded[5].

The Web application mainly focuses on: (i) semantic recommendation of papers; (ii) novel semantic search & retrieval of papers; (iii) data interlinking of bibliographical data with related data sources from LOD; (iv) innovative user interface design; and (v) sentiment analysis of extracted paper citations. Finally, our Web application also provides an in-depth analysis of the data that guides a user on his research field.

[1] `http://dl.acm.org`
[2] `http://www.elsevier.com`
[3] `http://www.acm.org/about/class/`
[4] The LAK challenge is an initiative of SOLAR (`http://www.solaresearch.org`) and LinkedUp project (`http://linkedup-project.eu/`).
[5] `http://lak.linkededucation.org`

5.2 Cite4Me

Cite4Me implements semantic and co-occurrence-based methods to search and retrieve academic papers and suggest related work in a user-friendly interface that assists users in exploring relationships between authors, institutions, papers and query terms. In what follows, we present the most relevant features of *Cite4Me* to the Semantic Web field and related to this thesis.

5.2.1 Search and Retrieval

In this section, we provide an overview of the major features of the Web application and its Web interface. *Cite4Me* implements standard techniques, such as free text search, to search and retrieve scientific publications, as well as semantic and exploratory search mechanisms. The main features is described as follows:

Free Text Search

The purpose of the *free text search* functionality is to offer users the abilities to search for mentions, titles and authors of academic publications contained in the *LAK dataset*. Even though, this functionality is similar to existing digital libraries, we agree that this is a basic functionality that must be provided by our application. Therefore, we use standard vector space models (*tf-idf*) for indexing and retrieving documents.

The *tf-idf* scores are computed for each term extracted from the publication content after applying stemming [101]. Furthermore, the searching functionality offers boolean queries with standard operators, such as *'OR'*, *'AND'*, and also a ranking of the matching publications based on the sum of *tf-idf* scores from the individual query terms.

Concisely, our free text search provides to the users publications (P) that match query terms and non-matching publications P', which are related to P according to a degree of similarity (see Eq. 5-1), but does not contain the query terms.

The similarity between a matching publication P and other non-matching publication P' in the LAK dataset is measured by the standard *cosine similarity measure*, which is built on top of the computed *tf-idf* scores.

$$Sim(P, P') = \frac{P \cdot P'}{|P||P'|} \tag{5-1}$$

where P and P' represent the *tf-idf* scores for the terms in two distinct publications.

Exploratory Search

The *exploratory search* or *graph search* component assists users to discover related literature, people and institutions that are working on a specific topic. A crucial step to provide this type of search is the annotation of the publications' content. For this, we used DBpedia Spotlight API for extracting entities, entity types and their categories (see steps S1 and S2 of the processing chain introduced in Section 4.3). For instance, the categories of the extracted concepts are used to interlink publications through the topics they cover. In cases where two publications share the same category (`dcterms:subject` property), then a link between both publications is created. Figure 5.1 shows an example of topically related publications.

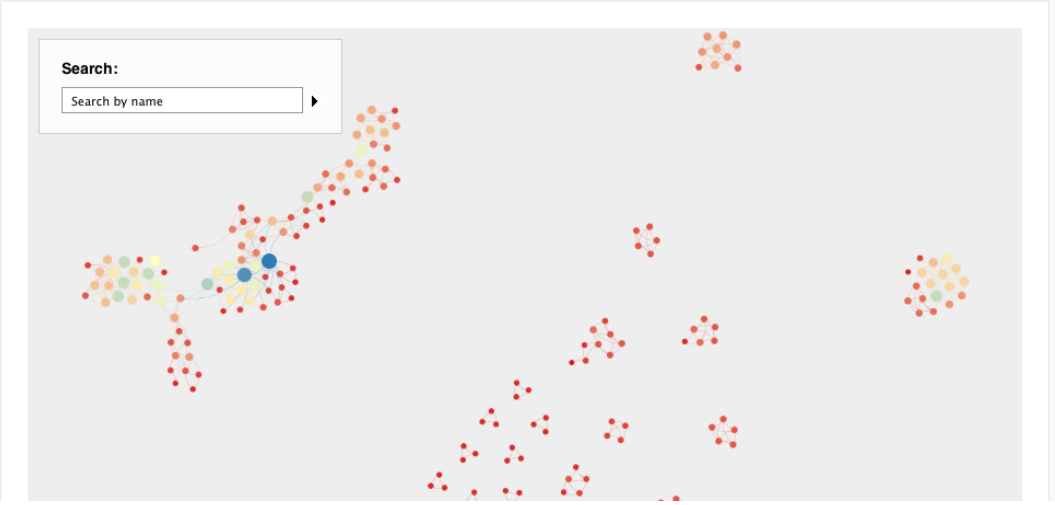


Figure 5.1: Preview of the exploratory search funcionality.

Semantic Search

After running the annotation process aforementioned, the relatedness score between the enriched concepts (i.e. DBpedia entities) found in the user query terms and the publications' content are computed and ranked. The relatedness score is computed based on the *tf-idf* score for the entities found in the publications' content. The ranking of the retrieved documents is based on the sum of the *tf-idf* scores of the matching concepts.

Figure 5.2 illustrates the semantic search functionality. Alongside the results of the semantic search a tag cloud shows the most prominent terms for a given user query. The tag cloud is updated while browsing through the list of results. The tags are selected based on the *tf-idf* score for the entities found in the abstract of the retrieved papers.

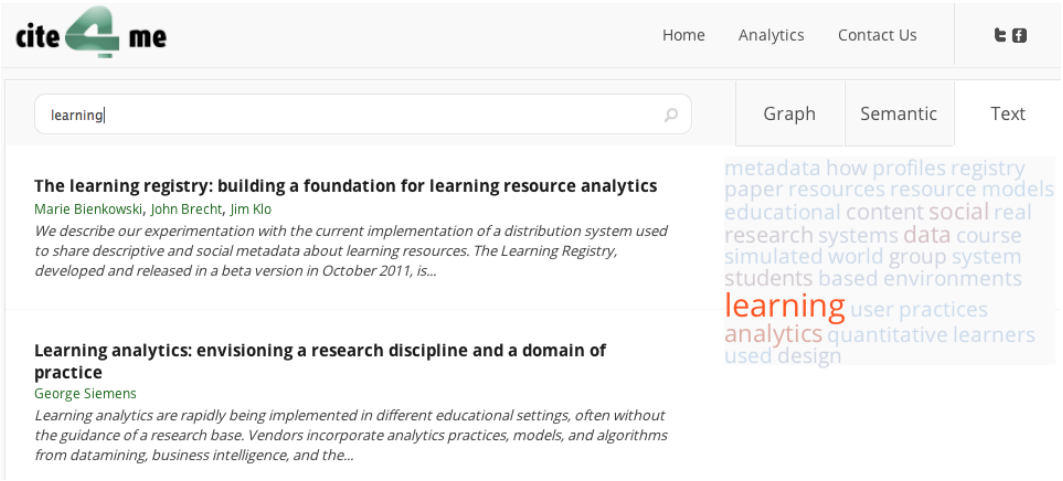


Figure 5.2: Preview of the semantic search funcionality.

Paper Recommendation

Another important feature of *Cite4Me* and which differentiates it from similar tools is the *semantic paper recommendation*. Given a scientific publication, the tool recommends a paper based on a score computed according to direct and lateral relationships between the publication of interest and the remaining papers in the corpus. To recommend publications, we rely on the approaches presented in Chapter 3 and 4.

Firstly, the paths connecting two enriched entities in the scientific publications are computed using the semantic connectivity score (SCS) (see Eq. 3-1). Next, the paper recommendation relies on an aggregated measure that takes into account the connectivity inter-documents (see Chapter 4 for more details). Finally, we generate a ranked list of pairwise publications according to the overall score, and the top-ranked publication is recommended to the user. Figure 5.3 illustrates the recommendation feature.

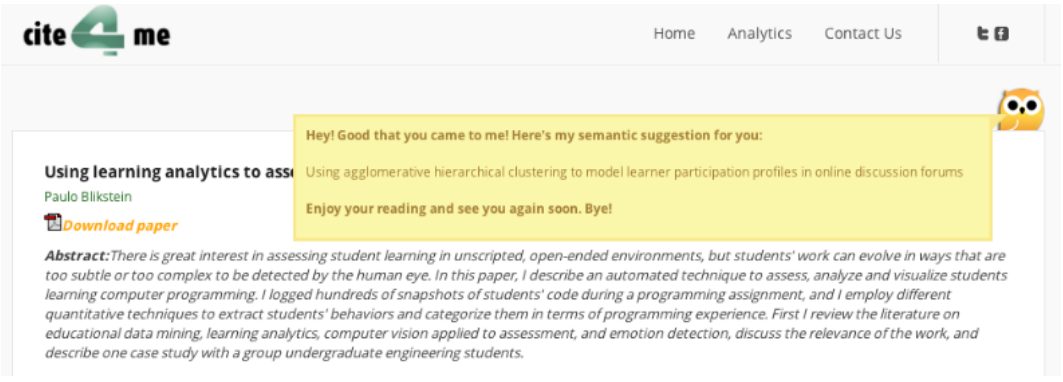


Figure 5.3: An example of paper recommendation based on SCS and SCS_d scores.

5.3 Conclusion

This chapter presented the application of previous works in the Semantic Web field within *Cite4Me*, a Web application that assists users in finding relevant scientific papers by exploring semantic relationships between them. *Cite4Me* is available at `http://www.cite4me.com`.

Currently, *Cite4Me* is linked to a dataset (*LAK Dataset*[6]) that contains semi-structured research publications from the ACM Digital Library (under a special license) and other public datasets (see also [102] for details). The dataset contains 315 full papers along with their descriptive metadata while new publications are added continuously. Metadata as well as the full text body are freely available in a variety of formats, including RDF accessible via a public SPARQL endpoint. We are currently working on expanding the number of papers available in *Cite4Me*. However, due to copyright reasons, the process to expose scientific publications from publishers is still under discussion.

[6] `http://www.solaresearch.org/resources/lak-dataset/`

6

Conclusions and Future Work

In this thesis we focused on the development of approaches that tackle data integration, consolidation and linkage problems posed by the rapid and heterogeneous growth of data on the Web over the years. To demonstrate the potential and usefulness of the proposed approaches, we also implemented a complete Web-based application and applied to a real-world scenario.

The first proposed approach addressed the longstanding and still-largely-open problem, namely determining complex (datatype property) mappings between ontologies. Such an approach is necessary and paramount towards data integration, mapping the heterogeneous data representation of similar concepts. That is, determining transformation rules between multiple datatype properties from a source ontology to a single property in the target ontology (e.g. *firstName* + *lastName* maps to *fullName*).

The problem of interest has been studied in a number of forms in previous literature [38, 39, 40, 41]. However, unlike most of the work in this area, our proposed algorithm does not only find 1:1 mappings, but is also capable of *automatically* identifying complex mappings.

Thus, we rely on a two-phase instance-based technique for complex datatype property matching (see Chapter 2). The first phase computes an estimated mutual information matrix to find simple mappings and suggest complex ones, whereas the second phase employs genetic programming to find complex matches from a reduced search space as a result of the first phase.

Empirical results show that our two-phase approach produces better results than when applied separately (see Section 2.6). Further improvement of the two-phased approach lies on the run time, with a reduction of approximately 36% in contrast to the run time of the genetic programming approach alone. Finally, in

terms of accuracy, the obtained results outperform those of previous state of the art approaches, such as iMap [36] and LSD [37]. The best performance was reported by iMap, where they obtained 0.84 and 0.55 for 1:1 and 1:n mappings, respectively, for the "Inventory" dataset. In contrast, we obtained an accuracy of 1 and 0.955 for 1:1 and 1:n, respectively.

In addition to the accurate results obtained, our approach can be directly extended to include additional transformation rules over known data types, such as date conversion functions, to find complex matches. We believe that adding popular transformation rules will help to increase performance of the genetic programming phase, since it will decrease the number of generations needed to find the correct transformation rule. Furthermore, we study to apply our approach to closely related problems, such as schema mapping evolution problem [103] and record deduplication [62].

Applying matching algorithms to map similar datatype properties between disparate ontologies provided a means to facilitate the process of data integration between disparate datasets. In Chapter 3, we presented a general-purpose approach for the discovery of relationships between entities, and further extended in Chapter 4 towards assessing document connectivity.

For the entity linking problem, we used the semantic-based entity connectivity approach (SCS), which is based on social network theory [70]. A comprehensive evaluation showed that SCS was able to uncover 14.3% entity connections not found by the co-occurrence measure CBM. Moreover, the semantic connectivity score for document (SCS_d) uncovered 16% of unique inferred document connections based on entity co-occurrence. Finally, an intuitive direction reflecting the completeness of uncovered latent entity connection was the combination of both approaches, which resulted in a $F1$ measure of 43% and 52% for entity and document connectivity, respectively.

Amongst the most important outcomes of the proposed SCS is its ability to reveal connections between entities and, hence, documents that linguistic and co-occurrence approaches would not reveal. The key advantage lies in the exploitation of semantic relations between entities in reference datasets (e.g. DBpedia).

The results also show that a combination of our semantic approach and traditional co-occurrence-based measures provided very promising results for detecting related entities as well as documents. While both approaches (CBM and SCS) produce fairly good indicators for entity and document connectivity, an evaluation based on Kendall's tau rank correlation showed that the approaches differ in the relationships they uncover.

As for future work, we aim at: (a) applying weights to different edge/property types according to their inherent semantics in order to provide a more refined score; (b) investigating means to combine our complementary approaches; and (c) applying our work to other, more domain-specific datasets.

In particular the latter step (c) will open opportunities to significantly improve results by tailoring our measure to specific node and edge types. Further research directions should focus on reducing the gathering of paths to a limited set of nodes ("hub nodes"), that is, due to the high computational demands when applying our approach to large-scale data such a selection of a limited set of nodes would improve scalability.

The last contribution in this thesis was presented in the form of a Web-based application. The goal of *Cite4Me* was to implement the proposed approaches and show the applicability to real-world scenarios. Due to its innovative features, such as semantic search, recommendation and graph visualisation, *Cite4Me* was awarded in the LAK challenge 2013.

Finally, attention ought to be drawn to some of the more serious social implications of this research. Whilst enabling the bringing forth of a better connected Web, this thesis also needs to address the complex issues of personal data privacy protection.

To address this issue, we conducted an experiment to verify the awareness of Web users regarding their privacy. The experiment was conducted on the developed Web-based platform, called FireMe[1]. There, we investigated the tweets of users and their posting behavior on Twitter[2]. The tweets were analysed to raise awareness of the consequences of such publicly available data with respect to the expression of negative sentiment of employees towards their bosses or jobs.

The results and the publicity gained with this experiment (see Appendix A) showed that most of the social network users are not aware of the possible harmful consequences of being exposed on the Web, either by posting or having public profile information.

Hence, as future work, we recommend to investigate solutions to the risks associated with the integration of users data with the Web of Linked Data. The integration of data from different sources might violate an individual's personal privacy and, therefore, harm them. A recent initiative of the Observatory for

[1] http://www.fireme.me
[2] http://twitter.com

Responsible Research Innovation[3] and the Website 'Please rob me'[4] also alert for the risks of having such personal data publicly published on the Web.

[3] http://responsible-innovation.org.uk/torrii/resource-detail/65
[4] http://www.pleaserobme.com/

7

Bibliography

[1] HU, W. et al. How matchable are four thousand ontologies on the semantic web. In: **Proceedings of the 8th Extended Semantic Web Conference on The Semantic Web: Research and Applications - Volume Part I**. Berlin, Heidelberg: Springer-Verlag, 2011. (ESWC'11), p. 290–304. ISBN 978-3-642-21033-4. Disponível em: <http://dl.acm.org/citation.cfm?id=2008892.2008918>. 1.1

[2] FERRARA, A.; NIKOLOV, A.; SCHARFFE, F. Data linking for the semantic web. **International Journal on Semantic Web and Information Systems**, v. 7, n. 3, p. 46–76, July/Summer 2011. Disponível em: <http://oro.open.ac.uk/33222/>. 1.1, 3.1

[3] HALPIN, H. et al. When owl: sameas isn't the same: an analysis of identity in linked data. In: **Proc. of the 9th International Semantic Web Conference,Vol. Part I**. Berlin, Heidelberg: [s.n.], 2010. p. 305–320. ISBN 3-642-17745-X, 978-3-642-17745-3. Disponível em: <http://dl.acm.org/citation.cfm?id=1940281.1940302>. 1.1, 3.1

[4] NUNES, B. P. et al. Complex matching of rdf datatype properties. In: **Ontology Matching**. [S.l.: s.n.], 2011. 1.3, 2.7

[5] NUNES, B. P. et al. Complex matching of rdf datatype properties. In: DECKER, H. et al. (Ed.). **Proceedings of Database and Expert Systems Applications - 24th International Conference, DEXA 2013, Prague, Czech Republic, August 26-29, 2013. Proceedings, Part I**. [S.l.]: Springer, 2013. (Lecture Notes in Computer Science, v. 8055), p. 195–208. ISBN 978-3-642-40284-5. 1.3, 2.7

[6] NUNES, B. P. et al. Can entities be friends? In: RIZZO, G. et al. (Ed.). **Proceedings of the Web of Linked Entities Workshop in conjuction**

with the 11th International Semantic Web Conference. [s.n.], 2012. (CEUR-WS.org, v. 906), p. 45–57. Disponível em: <http://ceur-ws.org/Vol-906/paper6.pdf>. 1.3, 3.3.1, 3.7

[7] NUNES, B. P. et al. Combining a co-occurrence-based and a semantic measure for entity linking. In: **The Semantic Web: Semantics and Big Data, 10th International Conference (ESWC)**. [S.l.: s.n.], 2013. p. 548–562. 1.3

[8] NUNES, B. P. et al. Interlinking documents based on semantic graphs. In: WATADA, J. et al. (Ed.). **17th International Conference in Knowledge Based and Intelligent Information and Engineering Systems, KES 2013, Kitakyushu, Japan, 9-11 September 2013**. [S.l.]: Elsevier, 2013. (Procedia Computer Science, v. 22), p. 231–240. 1.3

[9] NUNES, B. P.; FETAHU, B.; CASANOVA, M. A. Cite4me: Semantic retrieval and analysis of scientific publications. In: D'AQUIN, M. et al. (Ed.). **Proceedings of the LAK Data Challenge, Leuven, Belgium, April 9, 2013**. [S.l.]: CEUR-WS.org, 2013. (CEUR Workshop Proceedings, v. 974). 1.3

[10] NUNES, B. P. et al. Cite4me: A semantic search and retrieval web application for scientific publications. In: BLOMQVIST, E.; GROZA, T. (Ed.). **Proceedings of the ISWC 2013 Posters & Demonstrations Track, Sydney, Australia, October 23, 2013**. [S.l.]: CEUR-WS.org, 2013. (CEUR Workshop Proceedings, v. 1035), p. 25–28. 1.3

[11] KAWASE, R. et al. Who wants to get fired? In: DAVIS, H. C. et al. (Ed.). **Web Science 2013 (co-located with ECRC), WebSci '13, Paris, France, May 2-4, 2013**. [S.l.]: ACM, 2013. p. 191–194. ISBN 978-1-4503-1889-1. 1.3

[12] DIETZE, S. et al. Interlinking educational resources and the web of data - a survey of challenges and approaches. **Emerald Program: electronic Library and Information Systems**, v. 47, n. 1, 2013. 1.3

[13] NUNES, B. P. et al. Answering confucius: The reason why we complicate. In: LEO, D. H. et al. (Ed.). **Scaling up Learning for Sustained Impact - 8th European Conference, on Technology Enhanced Learning, EC-TEL 2013, Paphos, Cyprus, September 17-21, 2013. Proceedings**. [S.l.]: Springer, 2013. (Lecture Notes in Computer Science, v. 8095), p. 496–501. ISBN 978-3-642-40813-7. 1.3

[14] NUNES, B. P. et al. As simple as it gets - a sentence simplifier for different learning levels and contexts. In: **IEEE 13th International Conference on**

Advanced Learning Technologies, ICALT 2013, Beijing, China, July 15-18, 2013. [S.l.]: IEEE, 2013. p. 128–132. 1.3

[15] KAWASE, R. et al. Automatic competence leveling of learning objects. In: **IEEE 13th International Conference on Advanced Learning Technologies, ICALT 2013, Beijing, China, July 15-18, 2013**. [S.l.]: IEEE, 2013. p. 149–153. 1.3

[16] KAWASE, R.; NUNES, B. P.; SIEHNDEL, P. Content-based movie recommendation within learning contexts. In: **IEEE 13th International Conference on Advanced Learning Technologies, ICALT 2013, Beijing, China, July 15-18, 2013**. [S.l.]: IEEE, 2013. p. 171–173. 1.3

[17] FETAHU, B.; NUNES, B. P.; DIETZE, S. Summaries on the fly: Query-based extraction of structured knowledge from web documents. In: DANIEL, F.; DOLOG, P.; LI, Q. (Ed.). **ICWE**. [S.l.]: Springer, 2013. (Lecture Notes in Computer Science, v. 7977), p. 249–264. ISBN 978-3-642-39199-6. 1.3, 4.1

[18] LEME, L. A. P. P. et al. Identifying candidate datasets for data interlinking. In: DANIEL, F.; DOLOG, P.; LI, Q. (Ed.). **Web Engineering - 13th International Conference, ICWE 2013, Aalborg, Denmark, July 8-12, 2013. Proceedings**. [S.l.]: Springer, 2013. (Lecture Notes in Computer Science, v. 7977), p. 354–366. ISBN 978-3-642-39199-6. 1.3

[19] CARABALLO, A. A. M. et al. Trt - a tripleset recommendation tool. In: BLOMQVIST, E.; GROZA, T. (Ed.). **Proceedings of the ISWC 2013 Posters & Demonstrations Track, Sydney, Australia, October 23, 2013**. [S.l.]: CEUR-WS.org, 2013. (CEUR Workshop Proceedings, v. 1035), p. 105–108. 1.3

[20] FETAHU, B. et al. Generating structured profiles of linked data graphs. In: BLOMQVIST, E.; GROZA, T. (Ed.). **Proceedings of the ISWC 2013 Posters & Demonstrations Track, Sydney, Australia, October 23, 2013**. [S.l.]: CEUR-WS.org, 2013. (CEUR Workshop Proceedings, v. 1035), p. 113–116. 1.3

[21] KAWASE, R. et al. Automatic classification of documents in cold-start scenarios. In: CAMACHO, D.; AKERKAR, R.; RODRÍGUEZ-MORENO, M. D. (Ed.). **3rd International Conference on Web Intelligence, Mining and Semantics, WIMS '13, Madrid, Spain, June 12-14, 2013**. [S.l.]: ACM, 2013. p. 19. ISBN 978-1-4503-1850-1. 1.3

[22] LOPES, G. R. et al. Recommending tripleset interlinking through a social network approach. In: LIN, X. et al. (Ed.). **Web Information Systems Engineering - WISE 2013 - 14th International Conference, Nanjing, China, October 13-15, 2013, Proceedings, Part I**. [S.l.]: Springer, 2013. (Lecture Notes in Computer Science, v. 8180), p. 149–161. ISBN 978-3-642-41229-5. 1.3

[23] FETAHU, B.; NUNES, B. P.; DIETZE, S. Towards focused knowledge extraction: query-based extraction of structured summaries. In: CARR, L. et al. (Ed.). **22nd International World Wide Web Conference, WWW'13, Rio de Janeiro, Brazil, May 13-17, 2013, Companion Volume**. [S.l.]: International World Wide Web Conferences Steering Committee / ACM, 2013. p. 77–78. ISBN 978-1-4503-2038-2. 1.3

[24] KAWASE, R. et al. Openscout: harvesting business and management learning objects from the web of data. In: CARR, L. et al. (Ed.). **22nd International World Wide Web Conference, WWW'13, Rio de Janeiro, Brazil, May 13-17, 2013, Companion Volume**. [S.l.]: International World Wide Web Conferences Steering Committee / ACM, 2013. p. 445–450. ISBN 978-1-4503-2038-2. 1.3

[25] KAWASE, R. et al. Towards automatic competence assignment of learning objects. In: RAVENSCROFT, A. et al. (Ed.). **21st Century Learning for 21st Century Skills - 7th European Conference of Technology Enhanced Learning, EC-TEL 2012, Saarbrücken, Germany, September 18-21, 2012. Proceedings**. [S.l.]: Springer, 2012. (Lecture Notes in Computer Science, v. 7563), p. 401–406. ISBN 978-3-642-33262-3. 1.3

[26] NUNES, B. P. et al. Annotation tool for enhancing e-learning courses. In: POPESCU, E. et al. (Ed.). **Advances in Web-Based Learning - ICWL 2012 - 11th International Conference, Sinaia, Romania, September 2-4, 2012. Proceedings**. [S.l.]: Springer, 2012. (Lecture Notes in Computer Science, v. 7558), p. 51–60. ISBN 978-3-642-33641-6. 1.3

[27] NUNES, B. P. et al. Boosting retrieval of digital spoken content. In: GRAÑA, M. et al. (Ed.). **Knowledge Engineering, Machine Learning and Lattice Computing with Applications - 16th International Conference, KES 2012, San Sebastian, Spain, September 10-12, 2012, Revised Selected Papers**. [S.l.]: Springer, 2012. (Lecture Notes in Computer Science, v. 7828), p. 153–162. ISBN 978-3-642-37342-8. 1.3

[28] NUNES, B. P. et al. Automatically generating multilingual, semantically enhanced, descriptions of digital audio and video objects on the web. In: GRAÑA, M. et al. (Ed.). **Advances in Knowledge-Based and Intelligent Information and Engineering Systems - 16th Annual KES Conference, San Sebastian, Spain, 10-12 September 2012**. [S.l.]: IOS Press, 2012. (Frontiers in Artificial Intelligence and Applications, v. 243), p. 575–584. ISBN 978-1-61499-104-5. 1.3

[29] EUZENAT, J.; SHVAIKO, P. **Ontology matching**. [S.l.]: Springer, 2007. 1-333 p. 2.1

[30] LEME, L. A. P. P. et al. Instance-based owl schema matching. In: **ICEIS**. [S.l.: s.n.], 2009. p. 14–26. 2.1, 2.2.2, 2.3.1, 2.3.1, 2.4.1, 2.7

[31] LEME, L. A. P. P. et al. Matching object catalogues. **ISSE**, v. 4, n. 4, p. 315–328, 2008. 2.1, 2.7

[32] KOZA, J. R. **Genetic programming: on the programming of computers by means of natural selection**. Cambridge, MA, USA: MIT Press, 1992. ISBN 0-262-11170-5. 2.3.2

[33] NUNES, B. P. et al. **Complex matching of RDF datatype properties**. [S.l.], September 2011. 2.3.2

[34] COVER, T. M.; THOMAS, J. A. **Elements of information theory**. New York: Wiley, 1991. 2.3.2

[35] MEFFERT, K. **JGAP - Java Genetic Algorithms and Genetic Programming Package**. 2013. `http://http://jgap.sf.net/`. [Online; accessed 31-January-2013]. 2.4.2

[36] DHAMANKAR, R. et al. imap: discovering complex semantic matches between database schemas. In: **Proceedings of the 2004 ACM SIGMOD international conference on Management of data**. New York, NY, USA: ACM, 2004. (SIGMOD '04), p. 383–394. ISBN 1-58113-859-8. Disponível em: <http://doi.acm.org/10.1145/1007568.1007612>. 2.5.3, 6

[37] DOAN, A.; DOMINGOS, P.; LEVY, A. Learning Source Descriptions for Data Integration. In: ACM SIGMOD. **Proceedings of the Third International Workshop on the Web and Databases**. Dallas, TX, 2000. p. 81–86. 2.5.3, 2.6, 6

[38] DOAN, A.; HALEVY, A. Y. Semantic integration research in the database community: A brief survey. **AI Magazine**, v. 26, n. 1, p. 83–94, 2005. 2.7, 6

[39] KALFOGLOU, Y.; SCHORLEMMER, W. M. Ontology mapping: the state of the art. **Knowledge Eng. Review**, v. 18, n. 1, p. 1–31, 2003. 2.7, 6

[40] RAHM, E.; BERNSTEIN, P. A. A survey of approaches to automatic schema matching. **VLDB Journal**, v. 10, n. 4, p. 334–350, 2001. 2.7, 6

[41] SHVAIKO, P.; EUZENAT, J. A survey of schema-based matching approaches. **Journal of Data Semantics IV**, p. 146–171, 2005. 2.7, 6

[42] BERNSTEIN, P. A.; MADHAVAN, J.; RAHM, E. Generic schema matching, ten years later. **PVLDB**, v. 4, n. 11, p. 695–701, 2011. 2.7

[43] SHVAIKO, P.; EUZENAT, J. Ontology matching: State of the art and future challenges. **IEEE Trans. Knowl. Data Eng.**, v. 25, n. 1, p. 158–176, 2013. 2.7

[44] DUAN, S.; FOKOUE, A.; SRINIVAS, K. One size does not fit all: Customizing ontology alignment using user feedback. In: **International Semantic Web Conference**. [S.l.: s.n.], 2010. p. 177–192. 2.7

[45] RITZE, D.; PAULHEIM, H. Towards an automatic parameterization of ontology matching tools based on example mappings. In: **Ontology Matching**. [S.l.: s.n.], 2011. 2.7

[46] DHAMANKAR, R. et al. imap: Discovering complex mappings between database schemas. In: **SIGMOD Conference**. [S.l.: s.n.], 2004. p. 383–394. 2.7

[47] ALBAGLI, S.; BEN-ELIYAHU-ZOHARY, R.; SHIMONY, S. E. Markov network based ontology matching. **Journal of Computer and System Sciences**, v. 78, n. 1, p. 105–118, 2012. 2.7

[48] SPOHR, D.; HOLLINK, L.; CIMIANO, P. A machine learning approach to multilingual and cross-lingual ontology matching. In: **International Semantic Web Conference**. [S.l.: s.n.], 2011. p. 665–680. 2.7

[49] DUAN, S. et al. Instance-based matching of large ontologies using locality-sensitive hashing. In: **International Semantic Web Conference**. [S.l.: s.n.], 2012. p. 49–64. 2.7

[50] JIMÉNEZ-RUIZ, E.; GRAU, B. C. Logmap: Logic-based and scalable ontology matching. In: **International Semantic Web Conference**. [S.l.: s.n.], 2011. p. 273–288. 2.7

[51] WANG, P.; ZHOU, Y.; XU, B. Matching large ontologies based on reduction anchors. In: **IJCAI**. [S.l.: s.n.], 2011. p. 2343–2348. 2.7

[52] GIUNCHIGLIA, F.; AUTAYEU, A.; PANE, J. S-match: An open source framework for matching lightweight ontologies. **Semantic Web Journal**, v. 3, n. 3, p. 307–317, 2012. 2.7

[53] RAUNICH, S.; RAHM, E. Atom: Automatic target-driven ontology merging. In: **ICDE Conference**. [S.l.: s.n.], 2011. p. 1276–1279. 2.7

[54] LI, J. et al. Rimom: A dynamic multistrategy ontology alignment framework. **Knowledge and Data Engineering, IEEE Transactions on**, IEEE, v. 21, n. 8, p. 1218–1232, 2009. 2.7

[55] DO, H. hai; RAHM, E. Coma - a system for flexible combination of schema matching approaches. In: **In VLDB**. [S.l.: s.n.], 2002. p. 610–621. 2.7

[56] AUMUELLER, D. et al. Schema and ontology matching with coma++. In: **Proceedings of the ACM SIGMOD International Conference on Management of Data**. [S.l.]: ACM, 2005. p. 906–908. ISBN 1-59593-060-4. 2.7

[57] MASSMANN, S. et al. Evolution of the coma match system. In: **Ontology Matching**. [S.l.: s.n.], 2011. 2.7

[58] NAGY, M.; VARGAS-VERA, M.; STOLARSKI, P. Dssim results for oaei 2009. In: **Ontology Matching**. [S.l.: s.n.], 2009. 2.7

[59] HANIF, M. S.; AONO, M. An efficient and scalable algorithm for segmented alignment of ontologies of arbitrary size. **Journal of Web Semantics**, v. 7, n. 4, p. 344–356, 2009. 2.7

[60] CRUZ, I. F.; ANTONELLI, F. P.; STROE, C. Agreementmaker: Efficient matching for large real-world schemas and ontologies. **PVLDB**, v. 2, n. 2, p. 1586–1589, 2009. 2.7

[61] LAMBRIX, P.; TAN, H. Sambo - a system for aligning and merging biomedical ontologies. **Journal of Web Semantics**, v. 4, n. 3, p. 196–206, 2006. 2.7

[62] CARVALHO, M. G. de et al. A genetic programming approach to record deduplication. **IEEE Trans. Knowl. Data Eng.**, v. 24, n. 3, p. 399–412, 2012. 2.7, 6

[63] BIZER, C.; HEATH, T.; BERNERS-LEE, T. Linked data - the story so far. **International Journal on Semantic Web and Information Systems (IJSWIS)**, v. 5, n. 3, p. 1–22, 2009. 3.1

[64] SUCHANEK, F. M.; KASNECI, G.; WEIKUM, G. Yago: A core of semantic knowledge. In: **16th international World Wide Web conference**. New York, NY, USA: ACM Press, 2007. 3.1

[65] DIETZE, S. et al. Linked education: interlinking educational resources and the web of data. In: **Proceedings of the 27th ACM Symposium On Applied Computing (SAC-2012), Special Track on Semantic Web and Applications**. New York, NY, USA: ACM, 2012. (SAC '12). 3.1, 3.2

[66] CUNNINGHAM, H. et al. GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications. In: **Proc. of the 40th Anniversary Meeting of the Association for Computational Linguistics (ACL'02)**. Philadelphia: [s.n.], 2002. p. 168–175. 3.1

[67] RISSE, T. et al. Exploiting the social and semantic web for guided web archiving. In: **Proceedings of the Second international conference on Theory and Practice of Digital Libraries**. Berlin, Heidelberg: Springer-Verlag, 2012. (TPDL'12), p. 426–432. ISBN 978-3-642-33289-0. Disponível em: <http://dx.doi.org/10.1007/978-3-642-33290-6$_4$7>. 3.1

[68] ANYANWU, K.; SHETH, A. p-queries: enabling querying for semantic associations on the semantic web. In: **Proceedings of the 12th international conference on World Wide Web**. Budapest, Hungary: ACM Press New York, NY, USA, 2003. p. 690 – 699. 3.1

[69] SHETH, A. P.; RAMAKRISHNAN, C. Relationship web: Blazing semantic trails between web resources. **IEEE Internet Computing**, v. 11, n. 4, p. 77–81, 2007. 3.1

[70] KATZ, L.; KATZ, L. A new status index derived from sociometric analysis. **Psychometrika**, Springer New York, v. 18, n. 1, p. 39–43, mar. 1953. ISSN 0033-3123. Disponível em: <http://dx.doi.org/10.1007/bf02289026>. 3.1, 3.3.1, 6

[71] DIETZE, S. et al. Entity extraction and consolidation for social web content preservation. In: MITSCHICK, A. et al. (Ed.). **SDA**. [S.l.]: CEUR-WS.org, 2012. (CEUR Workshop Proceedings, v. 912), p. 18–29. 3.2

[72] DAMLJANOVIC, D.; STANKOVIC, M.; LAUBLET, P. Linked data-based concept recommendation: Comparison of different methods in open innovation

scenario. In: SIMPERL, E. et al. (Ed.). **The Semantic Web: Research and Applications - 9th Extended Semantic Web Conference, ESWC 2012, Heraklion, Crete, Greece, May 27-31, 2012. Proceedings**. [S.l.]: Springer, 2012. (LNCS, v. 7295), p. 24–38. ISBN 978-3-642-30283-1. 3.3.1, 3.6

[73] GRAVES, A.; ADALI, S.; HENDLER, J. A method to rank nodes in an rdf graph. In: BIZER, C.; JOSHI, A. (Ed.). **Proceedings of the Poster and Demonstration Session at the 7th International Semantic Web Conference (ISWC2008), Karlsruhe, Germany, October 28, 2008**. [S.l.]: CEUR-WS.org, 2008. (CEUR Workshop Proceedings, v. 401). 3.3.1

[74] FANG, L. et al. Rex: explaining relationships between entity pairs. **Proc. VLDB Endow.**, VLDB Endowment, v. 5, n. 3, p. 241–252, nov. 2011. ISSN 2150-8097. Disponível em: <http://dl.acm.org/citation.cfm?id=2078331.2078339>. 3.3.1, 3.6

[75] WATTS, D. J.; STROGATZ, S. H. Collective dynamics of 'small-world' networks. **Nature**, Nature Publishing Group, Department of Theoretical and Applied Mechanics, Cornell University, Ithaca, New York 14853, USA. djw24@columbia.edu, v. 393, n. 6684, p. 440–442, jun. 1998. ISSN 0028-0836. Disponível em: <http://dx.doi.org/10.1038/30918>. 3.3.1

[76] CHURCH, K. W.; HANKS, P. Word association norms, mutual information, and lexicography. **Comput. Linguist.**, MIT Press, Cambridge, MA, USA, v. 16, n. 1, p. 22–29, mar. 1990. ISSN 0891-2017. Disponível em: <http://dl.acm.org/citation.cfm?id=89086.89095>. 3.3.2

[77] GLIGOROV, R. et al. Using google distance to weight approximate ontology matches. In: **Proceedings of the 16th international conference on World Wide Web**. New York, NY, USA: ACM, 2007. (WWW '07), p. 767–776. ISBN 978-1-59593-654-7. Disponível em: <http://doi.acm.org/10.1145/1242572.1242676>. 3.3.2

[78] GABRILOVICH, E.; MARKOVITCH, S. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In: **Proceedings of the 20th international joint conference on Artifical intelligence**. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2007. (IJCAI'07), p. 1606–1611. Disponível em: <http://dl.acm.org/citation.cfm?id=1625275.1625535>. 3.4.3

[79] LEHMANN, J.; SCHUPPEL, J.; AUER, S. Discovering unknown connections - the dbpedia relationship finder. In: AUER, S. et al. (Ed.). **The Social Semantic Web 2007, Proceedings of the 1st Conference on Social Semantic**

Web (CSSW), September 26-28, 2007, Leipzig, Germany. [S.l.]: GI, 2007. (LNI, v. 113), p. 99–110. ISBN 978-3-88579-207-9. 3.6

[80] SEO, D. et al. Efficient finding relationship between individuals in a mass ontology database. In: KIM, T.-H. et al. (Ed.). **U- and E-Service, Science and Technology - International Conference, UNESST 2011, Held as Part of the Future Generation Information Technology Conference, FGIT 2011, in Conjunction with GDC 2011, Jeju Island, Korea, December 8-10, 2011. Proceedings**. [S.l.]: Springer, 2011. (CCIS, v. 264), p. 281–286. ISBN 978-3-642-27209-7. 3.6

[81] SABOU, M.; D'AQUIN, M.; MOTTA, E. Exploring the semantic web as background knowledge for ontology matching. **J. Data Semantics**, Springer, v. 11, p. 156–190, 2008. 3.6

[82] SABOU, M.; D'AQUIN, M.; MOTTA, E. Relation discovery from the semantic web. In: **7th International Semantic Web Conference (ISWC2008)**. [S.l.: s.n.], 2008. 3.6

[83] HAN, Y.-J. et al. Ranking entities similar to an entity for a given relationship. In: ZHANG, B.-T.; ORGUN, M. A. (Ed.). **PRICAI**. Springer, 2010. (Lecture Notes in Computer Science, v. 6230), p. 409–420. ISBN 978-3-642-15245-0. Disponível em: <http://dblp.uni-trier.de/db/conf/pricai/pricai2010.htmlHanPLPK10>. 3.6

[84] ANYANWU, K.; MADUKO, A.; SHETH, A. Semrank ranking complex relationship search results on the semantic web. In: **Proc. 14th Int. World Wide Web Conf.** Chiba, Japan: [s.n.], 2005. 3.6

[85] LESKOVEC, J.; HUTTENLOCHER, D.; KLEINBERG, J. Predicting positive and negative links in online social networks. In: **Proceedings of the 19th international conference on World wide web**. New York, NY, USA: ACM, 2010. (WWW '10), p. 641–650. ISBN 978-1-60558-799-8. Disponível em: <http://doi.acm.org/10.1145/1772690.1772756>. 3.6, 4.6

[86] GIONIS, A.; LAPPAS, T.; TERZI, E. Estimating entity importance via counting set covers. In: **KDD**. [S.l.: s.n.], 2012. p. 687–695. 3.6

[87] SIEMINSKI, A. Fast algorithm for assessing semantic similarity of texts. **Int. J. Intelligent Information and Database Systems**, v. 6, n. 5, p. 495–512, 2012. 3.6

[88] DUMAIS, S. T. Latent semantic analysis. **Annual Review of Information Science and Technology**, Wiley Subscription Services, Inc., A Wiley

Company, v. 38, n. 1, p. 188–230, 2004. ISSN 1550-8382. Disponível em: <http://dx.doi.org/10.1002/aris.1440380105>. 4.1

[89] MILNE, D.; WITTEN, I. H. An open-source toolkit for mining wikipedia. **Artificial Intelligence**, v. 194, n. 0, p. 222 – 239, 2013. ISSN 0004-3702. Disponível em: <http://www.sciencedirect.com/science/article/pii/S000437021200077X>. 4.3

[90] MENDES, P. N. et al. Dbpedia spotlight: shedding light on the web of documents. In: GHIDINI, C. et al. (Ed.). **Proceedings the 7th International Conference on Semantic Systems, I-SEMANTICS 2011, Graz, Austria, September 7-9, 2011**. [S.l.]: ACM, 2011. (ACM International Conference Proceeding Series), p. 1–8. ISBN 978-1-4503-0621-8. 4.3

[91] PASSANT, A. Measuring semantic distance on linking data and using it for resources recommendations. In: **Linked Data Meets Artificial Intelligence, Papers from the 2010 AAAI Spring Symposium, Technical Report SS-10-07, Stanford, California, USA, March 22-24, 2010**. [S.l.]: AAAI, 2010. 4.6

[92] PASSANT, A. dbrec - music recommendations using dbpedia. In: PATEL-SCHNEIDER, P. F. et al. (Ed.). **The Semantic Web - ISWC 2010 - 9th International Semantic Web Conference, ISWC 2010, Shanghai, China, November 7-11, 2010, Revised Selected Papers, Part II**. [S.l.]: Springer, 2010. (LNCS, v. 6497), p. 209–224. ISBN 978-3-642-17748-4. 4.6

[93] DEBNATH, S.; GANGULY, N.; MITRA, P. Feature weighting in content based recommendation system using social network analysis. In: **Proceedings of the 17th international conference on World Wide Web**. New York, NY, USA: ACM, 2008. (WWW '08), p. 1041–1042. ISBN 978-1-60558-085-2. Disponível em: <http://doi.acm.org/10.1145/1367497.1367646>. 4.6

[94] KALDOUDI, E.; DOVROLIS, N.; DIETZE, S. Information organization on the internet based on heterogeneous social networks. In: **Proceedings of the 29th ACM international conference on Design of communication**. New York, NY, USA: ACM, 2011. (SIGDOC '11), p. 107–114. ISBN 978-1-4503-0936-3. Disponível em: <http://doi.acm.org/10.1145/2038476.2038496>. 4.6

[95] THOR, A. et al. Link prediction for annotation graphs using graph summarization. In: **10th International Conference on The Semantic Web, Vol. Part I**. Berlin, Heidelberg: [s.n.], 2011.

(ISWC'11), p. 714–729. ISBN 978-3-642-25072-9. Disponível em: <http://dl.acm.org/citation.cfm?id=2063016.2063062>. 4.6

[96] POTAMIAS, M. et al. k-nearest neighbors in uncertain graphs. **Proc. VLDB Endow.**, VLDB Endowment, v. 3, n. 1-2, p. 997–1008, set. 2010. ISSN 2150-8097. Disponível em: <http://dl.acm.org/citation.cfm?id=1920841.1920967>. 4.6

[97] HASAN, M. A.; ZAKI, M. J. A survey of link prediction in social networks. In: AGGARWAL, C. C. (Ed.). **Social Network Data Analytics**. Springer, 2011. p. 243–275. ISBN 978-1-4419-8461-6. Disponível em: <http://dblp.uni-trier.de/db/books/collections/Social2011.htmlHasanZ11>. 4.6

[98] GROSS, A. et al. Mapping Composition for Matching Large Life Science Ontologies. In: **Proceedings of the 2nd International Conference on Biomedical Ontology**. [S.l.: s.n.], 2011. (ICBO 2011). 4.6

[99] VIDAL, V. M. P. et al. Query processing in a mediator based framework for linked data integration. **IJBDCN**, v. 7, n. 2, p. 29–47, 2011. Disponível em: <http://dblp.uni-trier.de/db/journals/ijbdcn/ijbdcn7.htmlVidalMPCP11>. 4.6

[100] XU, L.; EMBLEY, D. W. Discovering direct and indirect matches for schema elements. In: **DASFAA**. IEEE Computer Society, 2003. p. 39–46. Disponível em: <http://dblp.uni-trier.de/db/conf/dasfaa/dasfaa2003.htmlXuE03>. 4.6

[101] RIJSBERGEN, C. van; ROBERTSON, S.; PORTER, M. New models in probabilistic information retrieval. London: British Library. (British Library Research and Development Report, no. 5587), 1980. 5.2.1

[102] TAIBI, D.; DIETZE, S. Fostering analytics on learning analytics research: the lak dataset. In: **Proceedings of the LAK Data Challenge, held at LAK2013**. [S.l.: s.n.], 2013. 5.3

[103] FAGIN, R. et al. Schema mapping evolution through composition and inversion. In: BELLAHSENE, Z.; BONIFATI, A.; RAHM, E. (Ed.). **Schema Matching and Mapping**. Springer Berlin Heidelberg, 2011, (Data-Centric Systems and Applications). p. 191–222. ISBN 978-3-642-16517-7. Disponível em: <http://dx.doi.org/10.1007/978-3-642-16518-4$_7$>. 6

A

FireMe in the media

Since March 2013, FireMe has received a lot of attention from the press. Find below a few selected articles and video coverage worldwide (see Figure A.1).

Video coverage

Wall Street Journal Live: Will This Tweet Get You Fired? Ask FireMe! (English)
http://blogs.wsj.com/digits/2013/03/27/will-this-tweet-get-you-fired-ask-fireme/

FOX 5: FireMe! app posts your tweets about job, bosses (English)
http://www.ksla.com/story/21813553/fireme-app-posts-tweets-about-your-job-boss

CNN: Twitter tool saves you from yourself (English)
http://edition.cnn.com/video/?/video/tech/2013/03/28/exp-twitter-tool-amanpour.cnn

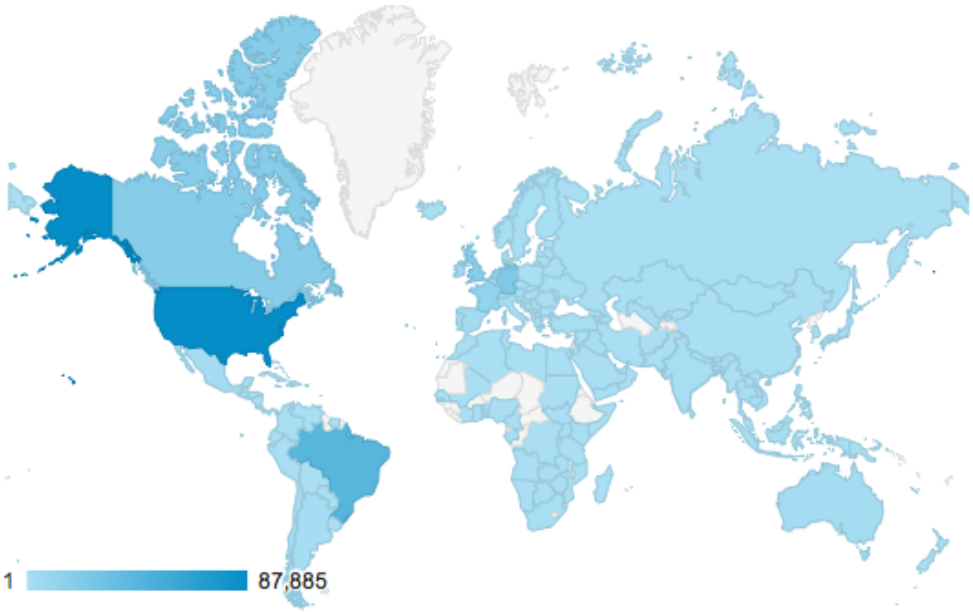


Figure A.1: Page views of FireMe app by country. FireMe was visited by Web users in 177 countries worldwide. (Traffic data taken from Google Analytics between March, 2013 and January, 2014)

EBC: Brasileiros criam site que identifica usuários do Twitter que falam mal do trabalho (Portuguese)
http://www.ebc.com.br/tecnologia/2013/05/brasileiros-criam-site-que-identifica-usuarios-do-twitter-que-falam-mal-do

CNN Chile: Fire Me: mide si está en riesgo tu trabajo por lo que dices en redes sociales (Spanish)
http://www.cnnchile.com/noticia/2013/03/27/fire-me-mide-si-esta-en-riesgo-tu-trabajo-por-lo-que-dices-en-redes-sociales

CNN México: Una herramienta analiza tuits negativos del trabajo
http://mexico.cnn.com/videos/2013/03/28/una-herramienta-analiza-tuits-negativos-del-trabajo

NDR: "Fire me": Wie User um Kündigung "betteln" (German)
http://www.ndr.de/ratgeber/netzwelt/fireme101.html

Articles in English

FireMe! App Tracks Boss-Hate On Twitter
http://www.huffingtonpost.com/2013/03/26/fireme-twitter-app_n_2955641.html

FireMe! Twitter alert says you've overstepped the mark
http://www.newscientist.com/blogs/onepercent/2013/03/fireme-twitter-alert.html

Will That Tweet Get You Fired? This App Warns You
http://mashable.com/2013/03/26/fire-me-app-twitter/

FireMe! Twitter Service Makes Getting Fired Way Easier
http://www.geekosystem.com/fire-me-twitter/

Calm Down, No One's Getting Fired Because Of FireMe!, New Site That Exposes People Tweeting Horrible Things About Their Jobs
http://techcrunch.com/2013/03/28/fireme/

Articles in German

Diese Tweets sollte Ihr Chef besser nicht lesen
http://www.bild.de/digital/internet/twitter/fire-me-diese-tweets-sollte-ihr-chef-besser-nicht-lesen-29740488.bild.html

Diese Website outet meckernde Job- und Chef-Hasser
http://www.krone.at/Digital/Diese_Website_outet_meckernde_Job-_und_Chef-Hasser-Fire_Me!-Story-356075

Twitter-App warnt User vor drohender Kündigung - "FireMe!" stöbert beleidigende Tweets gegen eigenen Chef auf
https://www.pressetext.com/#news/20130327018

Articles in Portuguese

Site rastreia profissionais que reclamam de seus chefes e empregos no Twitter - InfoMoney
http://www.infomoney.com.br/carreira/emprego/noticia/2713934/site-rastreia-profissionais-que-reclamam-seus-chefes-empregos-twitter

Nova ferramenta rastreia quem fala mal do trabalho no Twitter
http://www.techtudo.com.br/noticias/noticia/2013/03/nova-ferramenta-rastreia-quem-fala-mal-do-trabalho-no-twitter.html

Fire Me!: ferramenta rastreia quem fala mal do trabalho no Twitter
http://oglobo.globo.com/emprego/fire-me-ferramenta-rastreia-quem-fala-mal-do-trabalho-no-twitter-7950497

Articles in other languages

French (Le Monde)
http://bigbrowser.blog.lemonde.fr/2013/03/26/twitter-qui-veut-se-faire-virer/

Japanese (Irorio)
http://irorio.jp/asteroid-b-612/20130328/52165/

Macedonian (Kajgana)
http://kajgana.com/fireme-twitter-aplikacija-koja-gi-otkriva-hejterite

Italian (Dgmag)
http://www.dgmag.it/web/internet/fireme-38565-38565

Dutch (NRC)
http://www.nrc.nl/carriere/2013/04/08/kijk-hier-hoe-werknemers-op-twitter-massaal-hun-baas-afzeiken/