Effective Retrieval Model for Entity with Multi-Valued Attributes: BM25MF and Beyond*

Stéphane Campinas, Renaud Delbru, and Giovanni Tummarello

Digital Enterprise Research Institute
National University of Ireland, Galway
Galway, Ireland
firstname.lastname@deri.org

Abstract. The task of entity retrieval becomes increasingly prevalent as more and more structured information about entities is available on the Web in various forms such as documents embedding metadata (RDF, RDFa, Microdata, Microformats). International benchmarking campaigns, e.g., the Text REtrieval Conference or the Semantic Search Challenge, propose entity-oriented search tracks. This reflects the need for an effective search and discovery of entities. In this work, we present a multi-valued attributes model for entity retrieval which extends and generalises existing field-based ranking models. Our model introduces the concept of multi-valued attributes and enables attribute and value-specific normalization and weighting. Based on this model we extend two state-of-the-art field-based rankings, i.e., BM25F and PL2F, and demonstrate based on evaluations over heterogeneous datasets that this model improves significantly the retrieval performance compared to existing models. Finally, we introduce query dependent and independent weights specifically designed for our model which provide significant performance improvement.

Keywords: RDF, Entity Retrieval, Search, Ranking, Semi-Structured Data, BM25, BM25F, BM25MF, PL2, PL2F, PL2MF

1 Introduction

Despite the fact that the Web is best known as a large collection of textual documents, it also provides an increasing amount of structured data sources in various forms, from HTML tables to Deep Web databases, XML documents, documents embedding semantic markups, e.g., Microformats, Microdata, RDF, RDFa. Structured data on the Web covers a large range of domains, e.g., e-commerce, e-government, social network, scientific, editorial world, ..., and can describe any kind of *entities*, e.g., people, organisations, products, locations, etc.

Until now, search on the Web was mostly concerned with the retrieval of documents (i.e., unstructured text). However, the task of entity retrieval, that is, returning "entities"

*Preliminary results of the approach was presented in a technical report at SemSearch 2011 — http://semsearch.yahoo.com/9-Sindice.pdf. We have extended it with (1) an extension of the PL2F ranking function; (2) a study of optimised normalisation parameters, and (3) a comparison against two other field-based approaches over additional datasets.

in response to users' information needs, becomes increasingly prevalent as more and more structured information is available on the Web. This calls for systems providing effective means of searching and retrieving structured information. As mentioned in [1], searching over a large collection of structured data is an unsolved problem and is still an area in need of significant research. Entity retrieval has received considerable attention recently from various research communities [2, 3, 4, 5]. According to a recent study, more than half of web queries target a particular entity or instances of a given entity type [6]. Supporting effectively the search and retrieval of entities, therefore, is essential for ensuring a satisfying user experience.

One of the challenges in entity retrieval is to extend existing web search methods by exploiting the rich structure of the data. In this paper, we extend with a model that considers multi-valued attributes existing field-based ranking models, i.e., BM25F [7] and PL2F [8]. We define a multi-valued attribute as an attribute that has more than one value. For example, the email address of a person can be a multi-valued attribute since a person has possibly more than one. Our rationale for this new model is that field-based ranking models do not make a difference between single and multi-valued attributes. Usually, a multi-valued attribute is converted into a single-valued attribute by aggregating all the values into a single bag of words. Such a simplification of the underlying data model is inadequate for structured data and we will demonstrate that it can lead to a less effective ranking. In this paper, we introduce the MF ranking model which tackles specifically this problem, extend two popular ranking frameworks based on this model, and demonstrate its performance in comparison to existing field-based ranking models. Moreover, we introduce and evaluate additional extensions for combining attribute and value-specific weights.

In Section 2, we start by introducing our "Web of Data" model and then explain how we adapt existing field-based ranking models to this model. In Section 3 we review existing approaches for ranking entities in the Web of Data. In Section 4, we introduce the multi-valued attribute model MF and extend two state-of-the-art field-based models, i.e., BM25F and PL2F. Next, we present query dependent and independent weights specifically designed for our model. In Section 5, we compare field-based ranking models against their extension to the MF model on three large and heterogeneous datasets, and evaluate the effectiveness of the introduced weights.

2 Background

In this section, we first introduce a model for the Web of Data and define what is an entity, i.e., the unit of information that is queried and retrieved. Then we explain how to adapt two field-based ranking frameworks, i.e., the Probabilistic Relevance Framework [9] and the Divergence From Randomness [10], to this model.

2.1 The Web of Data

We define the *Web of Data* as the collection of structured data sources that are exposed on the Web through various forms such as HTML tables, Deep Web databases, XML documents, documents embedding semantic markups, e.g., Microformats, Microdata, RDF, RDFa. Since each data source might have its own defined schema, ranging from loosely to strictly defined, the data structure does not follow strict rules as in a database. Even within a given data source, the schema might not be fixed and may change as the information grows. The information structure evolves over time and new entities can require new attributes. We therefore consider the Web of Data as being *semi-structured* [11].

Web of Data Model In the rest of this paper, we assume that a common graph-based data model, based on the Resource Description Framework (RDF) model [12], is used for all the semi-structured data sources based on the Web. RDF is a generic data model that can be used for interoperable data exchange. A resource, i.e., an entity, is described in such a way that it can be processed by machines automatically. In RDF, a resource description is composed of statements about the resource. A statement is a triple consisting of a subject, a predicate and an object, and asserts that a subject has a property with some object. A set of RDF statements forms a directed labelled graph. In an RDF graph, as displayed in Figure 1, a node can be of three types: URI, literal and blank node. A URI serves as a globally-unique identifier for a resource. A literal is a character string with an optional associated language and datatype. A blank node represents a resource for which a URI is not given.

Entity Extraction There are multiple ways to extract entities from an RDF graph. Here, we use the approach described in [13], where we consider an entity as a star graph, i.e., a subgraph with a central node and its direct neighbor nodes it links to. Figure 1 displays how the RDF graph can be split into four entities *me*, *_:b1*, *_:b2* and *paper/547*. Each entity description forms a subgraph containing the incoming and outgoing edges of the entity node. In order to simplify the extraction process, we only consider the outgoing edges of a star graph.

Entity Model In the remainder of the paper, the unit of information which is retrieved and ranked is an *entity* [13] and is formalised as a list of attribute-value pairs:

Entity represents a set of attribute-value pairs and is identified by the entity node label, e.g., *paper/547*;

Attribute is an edge linking the entity node to one of its neighbor nodes and is identified by the edge label, e.g., *title*, *name* or *creator*;

Value is a neighbor node of the entity node and is identified by the node label, e.g., *Object-* or *paper/547*. A value is always associated to one attribute. Multiple values can be associated to a same attribute, such as the nodes *_:b1* and *_:b2* with the attribute *knows* of the entity *me*.

2.2 Field-Based Ranking Models

In this section, we explain how to adapt two existing field-based ranking frameworks to the entity model. In the Probabilistic Relevance Framework (PRF), BM25F [7] is a popular web-search ranking function for field-based document retrieval where a document is composed of multiple normalized weighted fields. For example, a field can be the title, the author or the body of the document. The Divergence From Randomness (DFR) framework gives birth to many ranking models, in particular PL2F [8] which considers

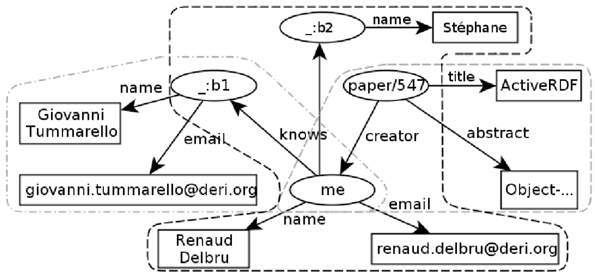


Fig. 1. An RDF graph divided into four entities identified by the nodes *me*, *_:b1*, *_:b2* and *paper/547*.

field-based documents similarly to BM25F. The mapping from the field-based document model to the entity model is straightforward. An entity can be seen as a document and an entity attribute as a document field. In presence of a multi-valued attribute, the common approach is to merge the content of all the values into one single value, i.e., creating a single bag of words.

Ranking Features The following features are used in the field-based ranking functions:

attribute length refers to the number of terms in a value node label. In case of a multi-valued attribute, it refers to the number of terms across all the values associated to the attribute.

average attribute length is equal to the mean of *attribute length* across entities.

BM25F Ranking Function Using BM25F, an entity e is scored with regard to a query q as follows:

$$Score(e,q) = \alpha_e \times \sum_{t \in q} q_t \times tfn \times \omega_t \tag{1}$$

$$tfn = \frac{f_{t,e} \times (k_1 + 1)}{f_{t,e} + k_1} \tag{2}$$

$$f_{t,e} = \sum_{a \in e} \frac{\alpha_a \times f_{t,e,a}}{1 + b_a \times \left(\frac{l_{e,a}}{l_a} - 1\right)} \tag{3}$$

where q_t is the weight of the query q for the term t, i.e., its frequency within the query q, tfn is the term frequency normalization function, ω_t is the Inverse Document Frequency (IDF) function of the term t, k_1 is the saturation parameter, $f_{t,e,a}$ is the frequency of the term t in the attribute a of the entity e, α_a is a weight of the attribute a and α_e a weight of the entity e, b_a is the normalization parameter for the attribute a with $b_a \in [0,1]$, $l_{e,a}$ is the *attribute length* of the attribute a in the entity e, l_a is the *average attribute length* of the attribute a. The IDF is defined as $\omega_t = 1 + log\left(\frac{N}{N_t+1}\right)$, where N is the total number of entities in the collection and N_t is the total number of entities that have occurrences of the term t.

PL2F Ranking Function DFR weighting models are based on the combination of three components, i.e., the information gain, the randomness model and the term frequency normalization model. PL2F bases the information gain on the Laplace after-effect model, uses Poisson as the model for randomness, and the *normalization 2F* for the term frequency normalization. Using PL2F, an entity e is scored with regard to a query q as follows:

$$\begin{aligned} Score(e,q) &= \alpha_e \times \sum_{t \in q} qtw \times w_{e,t} \\ w_{e,t} &= (1 - P_{risk}) \times (-log_2 (P_P)) \\ P_{risk} &= \frac{tfn}{1 + tfn} \\ P_P &= \frac{\lambda^{tfn}}{tfn!} \times e^{-\lambda} \text{ where } \lambda = \frac{TF}{N} \end{aligned}$$

where $qtw = \frac{q_t}{q_{t,max}}$ is the weight of the query q for the term t with $q_{t,max}$ the maximum of q_t in q, $w_{e,t}$ is the weight of the term t in the entity e, $1 - P_{risk}$ estimates the information gain of a term t, $-log_2 (P_P)$ evaluates the importance of a term t in the entity e thanks to the Poisson model and TF is equal to the frequency of the term t in the collection. The factorial is approximated with the Stirling's formula $tfn! = \sqrt{2\pi} \times tfn^{tfn+0.5} \times e^{-tfn}$. The term frequency of the term t in the entity e is normalized as follows:

$$tfn = \sum_{a \in e} \alpha_a \times f_{t,e,a} \times log_2 \left(1 + c_a \times \frac{l_a}{l_{e,a}} \right) \tag{4}$$

where c_a is a per-attribute hyperparameter with $c_a \in \left]0, +\infty\right[$.

3 Related Work

The Web of Data consists of a wide range of heterogeneous datasets, where the schema and the ontology can vary from one to the other. To overcome this diversity of attributes, different approaches for defining document fields have been proposed. In [14], the authors consider five weighted fields to represent the RDF structure of an entity: literals (textual values), keywords extracted from the entity label, i.e., the subject URI, keywords extracted from the incoming links, entity's types and keywords extracted from object URIs. Compared to the BM25F and PL2F approaches defined in 2.2, this approach is not able to grasp the rich structure of the data since attributes are completely discarded. The BM25F and PL2F approaches we use in our experiments are similar in nature to the BM25F adaptation proposed in [15], where the authors consider one field per attribute in the data collection and can assign a different weight to each attribute. However, they restrict their approach to attributes with literal values, discarding attributes with URI values. In contrast to [15], we consider both attributes with literal and URI values. URIs in the RDF graph carry relevant keywords, with regards to (1) the entity in general when considering the subject URI; (2) the attribute when considering the predicate URI; and (3) the resource the entity relates to when considering the object

URI. We also consider in our approach the entity and the attribute labels, i.e., the predicate URIs, using special entity attributes. This aspect is discussed in the Section 5.4.

However, all these approaches are an adaptation of the field-based ranking model in which multiple values associated to a same attribute are aggregated into a single value. This simplification of the underlying data model is inadequate for structured data as we will demonstrate in this paper. Therefore, we propose an extension of field-based ranking models in Section 4 to take into consideration multi-valued attributes and show that our model can be effectively applied to different ranking frameworks such as the PRF and the DFR.

4 MF Ranking Model

In this section we present the multi-valued attribute ranking model, denoted by "MF", which generalizes field-based ranking models to semi-structured data. The MF ranking model leads to a new definition of the term frequency normalization function that aggregates and normalises the term frequencies, first over all the values of an attribute, then over all the attributes of an entity. We introduce next the formal model then present two extensions to the MF model. Finally, we describe weights developed for the MF model.

Multi-Valued Attributes The MF model integrates multi-valued attributes with an additional intermediate computational step in the ranking function. Although values are related to a same attribute, the relevancy of each value with regard to the query is different. We can assume that, given a same attribute, an entity where two terms occur in a single value is more relevant than another entity where each term occurs in two values. This can be seen as a way to integrate some kind of term proximity measure in the retrieval model, i.e., two words are considered close to each other when they occur in the same value. Reflecting this difference into the importance of a term using an appropriate value-specific weight can improve the ranking efficiency. Therefore, we consider an attribute not as a bag of words but as a *bag of values*, each value being a *bag of words*.

Eliteness In [16], Harter introduced the notion of *eliteness* in order to model content-bearing terms: a document is said to be *elite* in term t if it is somehow "about" the topic associated with t. In [17], Robertson et al. introduce the relationship between the eliteness of a term in a document and its frequency: an elite term is most likely to be reused in the document, hence the term frequency is used as evidence of the term eliteness in the document. In [7], Zaragoza et al. extend the notion of eliteness to documents with multiple fields. In [18], the authors argue that the normalized frequencies of a term in each field should be combined before applying the term weighting model. Similarly in our MF model, the term eliteness in an entity is shared between its attributes. The values related to a same attribute are associated to a same topic, described by the attribute label. Therefore, a term eliteness in an attribute is shared between its values. As a consequence, for each term, we (1) accumulate the term's evidence of eliteness across an attribute's values; then (2) accumulate its evidence across the attributes; and finally (3) apply the term weighting model on the total term's evidence. The entity score is then derived by combination across the terms.

4.1 MF Ranking Functions

In this section, we describe *BM25MF* and *PL2MF*, the MF extensions of BM25F and PL2F, respectively. We present first the new features needed for the MF ranking model and then define both extensions.

Ranking Features The MF ranking model requires the following features:
value length refers to the number of terms in a value node label;
attribute length is equal to the *mean* of its *value length*;
average attribute length is the mean of *attribute length* across the entities where that attribute appears;
attribute cardinality is equal to the number of values an attribute possesses;
average attribute cardinality is equal to the mean of the *attribute cardinality* across the entities where that attribute appears.

BM25MF BM25F is extended by adapting the Equation (3) as follows:

$$f_{t,e} = \sum_{a \in e} \frac{\alpha_a \times f_{t,e,a}}{1 + b_a \times \left(\frac{|a|_e}{|a|} - 1\right)} \tag{5}$$

$$f_{t,e,a} = \sum_{v \in a} \frac{\alpha_v \times f_{t,e,v}}{1 + b_v \times \left(\frac{l_{e,v}}{l_a} - 1\right)} \tag{6}$$

where $f_{t,e,v}$ is the frequency of the term t within the value v in the entity e, $l_{e,v}$ is the *value length* of the value v in the entity e, $|a|_e$ is the *attribute cardinality* of the attribute a in the entity e, $|a|$ is the *average attribute cardinality* of the attribute a, α_v and α_a are respectively value and attribute specific weights, b_a and b_v are parameters of the term frequency's normalization, where b_v is value-specific and b_a attribute-specific with $(b_a, b_v) \in [0,1]^2$.

PL2MF PL2F is extended by adapting the Equation (4) as follows:

$$tfn = \sum_{a \in e} \alpha_a \times tfn_a \times log_2 \left(1 + c_a \times \frac{|a|}{|a|_e}\right) \tag{7}$$

$$tfn_a = \sum_{v \in a} \alpha_v \times f_{t,e,v} \times log_2 \left(1 + c_v \times \frac{l_a}{l_{e,v}}\right) \tag{8}$$

where c_a and c_v are hyperparameters, with c_a specific to the attribute a and c_v to the value v, with $(c_a, c_v) \in]0, +\infty[^2$.

In Equations (6) and (8), we normalize the term frequency based on the *average attribute length* l_a. In Equations (5) and (7), we further normalize the term frequency based on the *average attribute cardinality* $|a|$. In addition to attribute-specific weights α_a, the MF ranking model allows value-specific weights in its implementations with the parameter α_v. We will present value and attribute specific weights in the next section.

If we assume a single value per attribute to match field-based ranking models, then the Equations (6) and (5) are transformed into the Equation (3), with $\alpha_a \times \alpha_v$ as the BM25F's attribute weight, and b_v as the attribute normalization parameter. BM25MF is under this condition equivalent to BM25F. Under the same assumption, the Equations (8) and (7) are transformed into the Equation (4), with $\alpha_a \times \alpha_v \times log_2(1 + c_a)$ as the PL2F's attribute weight. PL2MF is under this condition equivalent to PL2F. Therefore, the MF model is a generalisation of field-based models for semi-structured data with multi-valued attributes.

4.2 Weights

In this section, we introduce several weights for the MF model. We first present two query-dependent weights: (1) the *Query Coverage* weight which indicates how well the query terms are covered by an entity, an attribute or a value; and (2) the *Value Coverage* weight which indicates how well a value node is covered by a query. Next, we describe the *Attribute and Entity Labels* query-independent weights.

The Query Coverage Weight The purpose of the Query Coverage (QC) weight is to lower the importance given to an entity, an attribute or a value with respect to the number of query terms it covers. This weight is combined with the ranking function using α_e, α_a and α_v. For example, given a query composed of three terms, if a value contains only occurrences of one query term, this value will then weight less than a value containing occurrences of more than one query term. It integrates the IDF weight of query terms so that the coverage takes into account the importance of the terms it covers. For example, if two entities have occurrences of one of the three query terms, the coverage would then be $\frac{1}{3}$ for both. Thanks to the IDF weights, the entity with the more important term will have a higher coverage weight than the other one. The QC weight is computed as $\frac{\sum_{t \in X \cap q} \omega_t^2}{\sum_{t \in q} \omega_t^2}$, where X is either a value, an attribute set or the entity and q is the query.

The Value Coverage Weight The Value Coverage (VC) weight reflects the proportion of terms in a value node matching the query, i.e., how much a query covers a value node. We assume that more the query terms match a large portion of the value node, the more this value node is a precise description of the query. This weight is combined with the ranking function using α_v. VC is defined as the quotient of the query terms frequencies in the value over the *value length*: $c' = \frac{\sum_{t \in v \cap q} f_{t,e,v}}{l_v}$. This definition disadvantages longer values over shorter ones: given a query with two terms, a small value with occurrences of one term would receive a higher weight than a larger value with the two terms occurring, because of the *value length* division. In order to have a better control over the effect of VC, we developed a function which (1) imposes a fixed lower bound to prevent short values receiving a higher weight than long ones; and (2) increases as a power function to ensure a high coverage weight only when c' is close to 1.

$$c_\alpha(c') = \frac{\alpha}{1 + (\alpha - 1) \times c'^B} \tag{9}$$

where $\alpha \in]0, 1[$ is a parameter that sets the lower bound of VC, and B is a parameter that controls the effect of the coverage on the value. The higher B is, the higher the coverage needs to be for the value node to receive a weight higher than α.

The Attribute and Entity Labels Weights The Attribute and Entity Labels (AEL) weights balance the importance of an entity or an attribute depending on its label. This weight is combined with the ranking function using α_a. The weight value is defined empirically. Comparing the label to a regular expression, the weight is equal (a) to 2 if the label matches "$. \star [label \,|\, name \,|\, title \,|\, sameas]\$$"; (b) to 0.5 if the label matches "$. \star [seealso \,|\, wikilinks]\$$"; and (c) to 0.1 if the label matches "`http://www.w3.org/1999/02/22-rdf-syntax-ns#_`$[0-9]+\$$"; (d) to 1 otherwise. For instance, if an attribute label is `http://xmlns.com/foaf/0.1/name` then a weight of 2 is assigned. The regular expression (c) matches an attribute URI defining items of a collection in RDF[1]. It is assigned a low weight of 0.1 to reduce the importance of terms occuring in each item of the collection. The entity label is treated as a special attribute of the entity and is assigned a weight of 2.

5 Experiments

In order to evaluate the MF model, we perform several experiments using three different datasets. We start by experimenting on the normalization parameters in order to study their impact on the effectiveness of the approach. We then compare the MF ranking functions against other traditional ones. We finally discuss the consequence of not considering the attribute label as a source of potential relevant terms and demonstrate the effectiveness of the proposed weights.

While there are other ways to perform entity ranking on the Web of Data, e.g., one can look at the other SemSearch 2011 candidates, in this evaluation we concentrate on demonstrating how the MF model specifically extends and improves the very popular PRF and DFR frameworks.

5.1 Datasets

The datasets we are using in our experiments are the following:

INEX09 a dataset of 2,491,134 triples from DBpedia containing the description of entities in English, and converted for the INEX evaluation framework [14];

SS10 the "Billion Triple Challenge"[2] (BTC) dataset, containing more than one billion triples, with the assessments of the SemSearch2010[3] challenge;

SS11 the BTC dataset with the assessments of the "Entity Search track" of the SemSearch2011[4] challenge.

The INEX09 dataset is significantly different than the other two based on BTC. Indeed, BTC is a heterogeneous dataset, created from web crawls of several search engines. INEX09 is a highly curated dataset from DBpedia.

5.2 Effectiveness of the MF Ranking Model

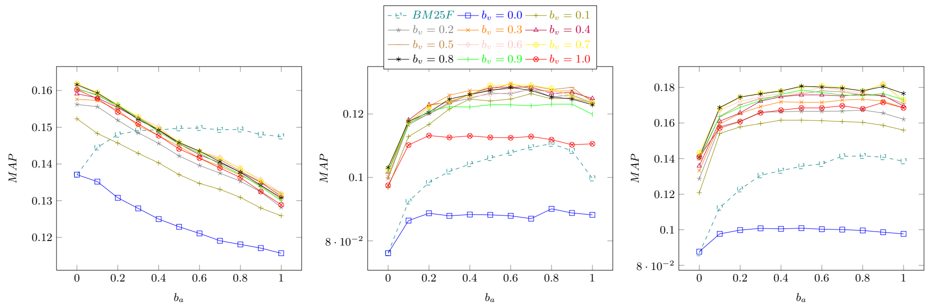
In this section, we study the impact on the retrieval performance of the normalization parameters. Next we perform a comparison between the MF extensions, i.e., BM25MF and PL2MF, against other ranking approaches.

[1] RDF Container: `http://www.w3.org/TR/rdf-schema/#ch_container`

[2] Billion Triple Challenge: `http://vmlion25.deri.ie`

[3] SemSearch2010: `http://km.aifb.kit.edu/ws/semsearch10`

[4] SemSearch2011: `http://km.aifb.kit.edu/ws/semsearch11`



(a) INEX09 (b) SS10 (c) SS11

Fig. 2. Experiment with the BM25MF normalization parameters. The figures report the MAP values of the respective datasets, where a curve plots a fixed b_v value with b_a varying from 0 to 1 with a precision step of 0.1.

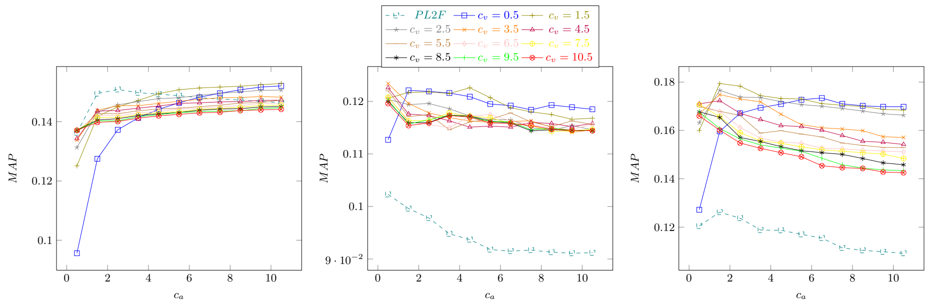
The Normalization Parameters The effectiveness of the methods from the PRF and DFR frameworks depends on finding the right values for the normalization parameters. However, these parameters are highly dependent on the dataset. In addition to the length normalization of field-based ranking function, the MF ranking function offers an additional normalization on the attribute's cardinality. The Figures 2 and 3 depict the impact of the normalization parameters on the retrieval performance of BM25MF and PL2MF respectively. Each figure depicts the Mean Average Precision (MAP) scores on the three datasets for BM25MF (resp., PL2MF), with the value normalization parameter b_v (resp., c_v) on the x axis and the MAP score on the y axis. Each curve plots the results with a fixed attribute's normalization parameter b_a (resp., c_a). The grid of parameters values in Figure 2 ranges from 0 to 1 with a step of 0.1. In Figure 3, the grid ranges from 0.5 to 10.5 with a step of 1. Although these parameters can be attribute and value-specific, this experiment considers a constant parameter in order to reduce the number of combinations and to lower the variability of the results. Dashed lines depict the MAP scores of BM25F and PL2F and solid lines the scores of their MF extension, BM25MF and PL2MF respectively. These plots show that using a normalization on the value node provides improved performance. Indeed, the attribute normalization parameters b_a and c_a alone do not grasp the heterogeneity in the data,which results in lower performance when compared to the MF extensions. This indicates that the distinction of an attribute being a set of values has a positive effect on the ranking.

5.3 Comparison between MF and Field-Based Models

In this section, we evaluate and compare the performance of BM25 and PL2 ranking model against their MF extensions, BM25MF and PL2F respectively, and show the superiority of the MF model. TF-IDF is used as baseline.

TF-IDF is a logarithmic function of the term frequency and defines the Equation (2) as $tfn = log(f_{t,e}) + 1$, where $f_{t,e}$ is the number of occurrences of the term t in the entity e.

BM25 [19] considers the document as a simple bag of words. It is a function of the term frequency derived from a two-Poisson model and using an entity-length normalization. The entity length is computed as the sum of the *attribute length* defined



(a) INEX09 (b) SS10 (c) SS11

Fig. 3. Experiment with the PL2MF normalization parameters. The figures report the MAP values of the respective datasets, where a curve plots a fixed c_v value with c_a varying from 0.5 to 10.5 with a precision step of 1.

in the Section 2.2. It defines the Equation (2) as $tfn = \frac{f_{t,e} \times (k_1+1)}{f_{t,e}+k_1 \times \left(1+b \times \left(\frac{l_e}{l_{avg}}-1\right)\right)}$, where l_e is the *entity length* of the entity e, l_{avg} is the average of the *entity length* in the collection and b is a normalization parameter.

BM25F is defined in Equation (2). It considers documents as composed of fields, each field being a bag of words.

PL2 [10] considers the document as a simple bag of words. It is a model derived from the DFR framework, with the Equation (4) formulated as $tfn = f_{t,e} \times log_2\left(1 + c \times \frac{l_{avg}}{l_e}\right)$, where c is a normalization parameter.

PL2F is defined in Equation (4). It considers a document as a set of fields, each field being a bag of words.

Comparison The Table 1a reports the values of the normalization parameters of each ranking function found through a constrained particle swarm optimization [20] on each dataset. Using such parameters, we report in Table 1b the performance of the ranking functions on the three datasets. The p-Value is computed with the two-tailed Wilcoxon matched-pairs signed-ranks test [21, 22], where a statistically significant difference at level 0.10 is marked with one star $*$ and at level 0.05 with two stars $**$. BM25MF and PL2MF are used as a baseline in this test. $\Delta\%$ indicates the difference in percentage between the two MAP values compared in that test. We note that for field-based ranking models and their MF extensions, the attribute label is considered as a value node, in order to be a source of potential relevant terms.

TF-IDF provides a clear-cut discrepancy between INEX09 and the datasets based on BTC, i.e., SS10 and SS11, the reason being it is not suited to heterogeneous datasets. On SS10, BM25MF (resp., PL2MF) does not report a significant difference with BM25 (resp., PL2). On INEX09 and SS11, the MF extensions provide an increase of at least 10% in retrieval performance compared to BM25 and PL2. On SS10 and SS11, the MF extensions provide better retrieval performance with a significant difference than the field-based ranking functions with an increase of 15% at the minimum. On INEX09, BM25MF provides slightly better results than BM25F. Overall, the experiments show that the MF model improves significantly the ranking effectiveness.

Table 1. Comparison of state-of-the-art candidates against the MF generalizations.

(a) Normalization parameters values, found through a constrained particle swarm optimization.

	INEX09		SS10		SS11	
BM25MF	$b_a = 0.00$	$b_v = 0.75$	$b_a = 0.58$	$b_v = 0.75$	$b_a = 0.58$	$b_v = 0.75$
BM25	$b = 0.20$		$b = 0.20$		$b = 0.20$	
BM25F	$b_a = 0.82$		$b_a = 0.82$		$b_a = 0.82$	
PL2MF	$c_a = 9.19$	$c_v = 0.76$	$c_a = 1.52$	$c_v = 1.03$	$c_a = 1.79$	$c_v = 1.88$
PL2	$c = 17.01$		$c = 10.09$		$c = 10.09$	
PL2F	$c_a = 1.87$		$c_a = 0.51$		$c_a = 1.51$	

(b) Mean Average Precision (MAP) and the Precision at 10 (P@10) scores of PL2MF and BM25MF and the other state-of-the-art candidates; a p-Value is computed using the two-tailed Wilcoxon matched-pairs signed-ranks test, where one star $*$ marks statistically significant difference at level 0.10, and two stars $**$ at level 0.05, with BM25MF (resp., PL2MF) used as a baseline; $\Delta\%$ indicates the difference in percentage between the two MAP values compared in that test.

	INEX09				SS10				SS11			
	MAP	P@10	p-Value	$\Delta\%$	MAP	P@10	p-Value	$\Delta\%$	MAP	P@10	p-Value	$\Delta\%$
BM25MF	0.1593	0.3982	-	-	0.1303	0.3783	-	-	0.1811	0.1880	-	-
TF-IDF	0.1246	0.3109	$3.2e^{-04**}$	-21.78	0.0581	0.2304	$2.1e^{-10**}$	-55.41	0.0655	0.1040	$1.4e^{-06**}$	-176.49
BM25	0.1330	0.3309	$8.4e^{-05**}$	-16.51	0.1350	0.4000	$3.7e^{-01}$	-	0.1625	0.1920	$1.4e^{-01*}$	-11.45
BM25F	0.1489	0.3764	$5.7e^{-03**}$	-6.53	0.1100	0.3283	$3.0e^{-05**}$	-15.58	0.1401	0.1680	$1.1e^{-04**}$	-29.26
PL2MF	0.1525	0.3800	-	-	0.1232	0.3707	-	-	0.1797	0.1880	-	-
TF-IDF	0.1246	0.3109	$5.4e^{-03**}$	-18.30	0.0581	0.2304	$5.4e^{-10**}$	-52.84	0.0655	0.1040	$6.9e^{-07**}$	-174.35
PL2	0.1331	0.3218	$6.0e^{-03**}$	-12.72	0.1289	0.3946	$4.1e^{-01}$	-	0.1614	0.2000	$3.3e^{-01*}$	-11.34
PL2F	0.1514	0.3473	$7.7e^{-01}$	-	0.1023	0.3163	$2.4e^{-04**}$	-16.96	0.1264	0.1560	$4.5e^{-05**}$	-42.17

5.4 Effectiveness of the Weights

In this section, we discuss the impact of discarding the attribute label as a source of possible relevant terms on the ranking performance. Then we evaluate the weights from Section 4.2 developed for the MF model. The Table 2 reports the MAP scores of BM25MF and PL2MF combined with each weight individually and using the normalization parameters values from the Table 1a. Apart from the row "Without Attribute Label", all runs consider the attribute label as an additional value as in the previous experiments.

The Impact of Attribute Label In this section, we investigate the consequence of not considering the attribute label as a source of relevant terms. The Table 2 reports under the *BM25MF* and *PL2MF* methods the results of considering or not the attribute label as an additional value. We can see that removing the attribute label (*Without Attribute Label* row) lowers the performance of the ranking with a statistical significance on INEX09 with BM25MF and PL2MF, and on SS11 with PL2MF only. This shows that the attribute labels can be a source of possible relevant terms.

The Query Coverage Weight In order to evaluate the benefit of the QC weight, we first analyse its effect separately when applied as an entity, an attribute or a value-specific weight. Then we study the consequence of applying it on all nodes at the same time (*All* row). The Table 2 reports the results under the *BM25MF + QC* and *PL2MF + QC* methods. QC improves the retrieval performance when applied on the attribute node, with a statistical significance on SS10 and SS11.

The Value Coverage Weight The evaluation of the VC weight investigates its efficiency with and without the function (9). The results are reported in Table 2 under the *BM25MF + VC* and *PL2MF + VC* methods. We provide for each dataset the best performing B and α parameters. We can observe that VC with the function (9) improves slightly the retrieval performance on SS10 and SS11. The reason is that, without this function, VC assigns a low weight to long values even if they have occurrences of all query terms.

The Attribute and Entity Labels Weights We evaluate the AEL weights first by considering the Attribute and the Entity Label weights separately, then both at the same time. The Table 2 reports the results of applying such query-independent weights under the *BM25MF + AEL* and *PL2MF + AEL* methods. We note that the Attribute Label weight gives significant benefits to the ranking in SS10 and SS11, while it decreases the ranking performance in INEX09. This indicates that carefully defined weights for important and non-important attributes can contribute significantly to the effectiveness of the approach. We note also that the same can be seen with the Entity Label weight applied alone. The reason is similar to the Attribute Label weight. Except in INEX09, using both weights at the same time increases the performance of MF ranking functions noticeably.

The Combination of Weights In this section, we investigate the retrieval performance when all four weights are used together. We report the results in Table 2 under the methods *BM25MF + AC + VC + AEL* and *PL2MF + AC + VC + AEL*, with the weights configuration (1) QC applied on the attribute node; (2) VC with dataset-specific B and α parameters; and (3) AEL weights. The weights applied on a same node are combined by the multiplication of each weight value on that node, i.e., either b_a (resp., c_a) or b_v (resp., c_v) weight values. The attribute label being considered as a value, and the entity label being an additional attribute, we apply also the QC weight on those two labels in this experiment. On INEX09 their combination decreases slightly the performance for PL2MF. On SS10 and SS11, although the QC and VC weights applied separately do not improve the effectiveness of the MF ranking functions by much, their combination with AEL increases the retrieval performance by at least 30% on SS10 and SS11.

6 Conclusion

In this paper, we introduced the MF model for ad-hoc retrieval of semi-structured data with multi-valued attributes. We have discussed how this model extends and generalises existing field-based models. Based on our MF model, we have developed two extensions BM25MF and PL2MF of two popular field-based ranking models, respectively BM25F and PL2F. We have shown throughout evaluations on three large and heterogeneous datasets that the MF model provides significant performance improvement over the field-based ranking models. In addition, the proposed approach provides additional extensions for combining attribute and value-specific weights. We have presented query dependent and independent weights specifically designed for our model and have demonstrated that such weights improve the overall performance.

We have shown in the evaluations that normalization parameters in the MF model are highly dependent on the datasets. A dynamic approach for finding such parameters

Table 2. Evaluation of the weights effectiveness on PL2MF and BM25MF.

Method	INEX09				SS10				SS11			
	MAP	P@10	p-Value	$\Delta\%$	MAP	P@10	p-Value	$\Delta\%$	MAP	P@10	p-Value	$\Delta\%$
BM25MF												
With Attribute Label	0.1593	0.3982	-	-	0.1303	0.3783	-	-	0.1811	0.1880	-	-
Without Attribute Label	0.1484	0.3800	$6.9e^{-04**}$	-6.84	0.1241	0.3783	$4.5e-01$	-	0.1763	0.1940	$8.6e-01$	-
BM25MF + QC												
Value	0.1482	0.3545	$2.0e^{-01*}$	-6.97	0.1325	0.3793	$8.2e^{-01}$	-	0.1841	0.2020	$1.6e^{-01*}$	$+1.66$
Attribute	0.1624	0.3818	$8.8e^{-01}$	-	0.1339	0.3815	$3.5e^{-01*}$	$+2.76$	0.1841	0.2060	$5.3e^{-02*}$	$+1.66$
Entity	0.1514	0.3782	$4.0e^{-02*}$	-4.96	0.1236	0.3728	$2.8e^{-02**}$	-5.14	0.1744	0.188	$4.7e^{-01}$	-
All	0.1506	0.3545	$1.5e^{-01*}$	-5.46	0.1263	0.3717	$3.1e^{-01*}$	-3.07	0.1810	0.2080	$7.1e^{-01}$	-
BM25MF + VC												
With Function (9)	0.1606	0.3964	$2.8e^{-01*}$	$+0.82$	0.1321	0.3761	$3.7e^{-01}$	-	0.1802	0.1920	$5.4e^{-01}$	-
		$B=1,\ \alpha=0.7$				$B=2,\ \alpha=0.4$				$B=1,\ \alpha=0.9$		
Without Function (9)	0.1293	0.3055	$8.9e^{-04**}$	-18.83	0.1260	0.3609	$5.3e^{-01}$	-	0.1296	0.1820	$2.1e^{-03**}$	-39.74
BM25MF + AEL												
Entity Label Weight	0.1574	0.3909	$9.1e^{-02*}$	-1.19	0.1401	0.4011	$7.8e^{-05**}$	$+7.52$	0.1937	0.2000	$6.1e^{-03**}$	$+6.96$
Attribute Label Weight	0.1604	0.3982	$4.3e^{-01}$	-	0.1504	0.4228	$2.6e^{-06**}$	$+15.43$	0.2173	0.2360	$6.8e^{-06**}$	$+19.99$
Both	0.1593	0.3982	$4.4e^{-01}$	-	0.1584	0.4391	$2.0e^{-07**}$	$+21.57$	0.2274	0.2420	$2.7e^{-05**}$	$+25.57$
BM25MF + AC + VC + AEL	0.1589		$5.9e^{-01}$	-	0.1720	0.4620	$3.8e^{-06**}$	$+32.00$	0.2416	0.2560	$1.1e^{-05**}$	$+33.41$
PL2MF												
With Attribute Label	0.1525	0.3800	-	-	0.1232	0.3685	-	-	0.1797	0.1880	-	-
Without Attribute Label	0.1401	0.3618	$2.3e^{-04**}$	-8.13	0.1192	0.3674	$5.3e^{-01}$	-	0.1680	0.1840	$1.6e^{-01*}$	-6.51
PL2MF + QC												
Value	0.1348	0.3382	$1.0e^{-03**}$	-11.61	0.1276	0.3750	$2.8e^{-01*}$	$+3.57$	0.1818	0.1980	$1.6e^{-01*}$	$+1.17$
Attribute	0.1569	0.3793	$8.4e^{-01}$	-	0.1299	0.3761	$1.6e^{-01*}$	$+5.44$	0.1815	0.1960	$6.5e^{-02*}$	$+1.00$
Entity	0.1499	0.3727	$3.9e^{-01}$	-	0.1168	0.3609	$3.8e^{-02**}$	-5.19	0.1655	0.1900	$1.2e^{-01*}$	-7.90
All	0.1374	0.3309	$1.5e^{-02**}$	-9.90	0.1257	0.3728	$6.4e^{-01}$	-	0.1743	0.2020	$9.4e^{-01}$	-3.01
PL2MF + VC												
With Function (9)	0.1494	0.3673	$1.5e^{-02**}$	-2.03	0.1253	0.3663	$2.4e^{-01*}$	$+1.70$	0.1802	0.1900	$1.4e^{-01*}$	$+0.29$
		$B=1,\ \alpha=0.7$				$B=2,\ \alpha=0.4$				$B=1,\ \alpha=0.9$		
Without Function (9)	0.1182	0.2909	$5.4e^{-07**}$	-21.84	0.1179	0.3391	$3.8e^{-01}$	-	0.1466	0.1900	$7.3e^{-03**}$	-18.42
PL2MF + AEL												
Entity Label Weight	0.1523	0.3800	$2.5e^{-01*}$	-0.13	0.1305	0.3848	$6.4e^{-06**}$	$+5.93$	0.1824	0.1920	$4.4e^{-02*}$	$+1.50$
Attribute Label Weight	0.1521	0.3673	$1.9e^{-02**}$	-0.26	0.1471	0.4163	$3.7e^{-07**}$	$+19.40$	0.2150	0.2320	$4.0e^{-05**}$	$+19.64$
Both	0.1516	0.3709	$1.2e^{-02**}$	-0.59	0.1542	0.4337	$3.4e^{-09**}$	$+25.16$	0.2187	0.2380	$1.2e^{-05**}$	$+21.70$
PL2MF + QC + VC + AEL	0.1492	0.3582	$1.5e^{-01*}$	-2.16	0.1717	0.4620	$2.0e^{-08**}$	$+39.36$	0.2360	0.2520	$2.5e^{-05**}$	$+31.33$

would improve the performance of the approach and will be investigated in a future work. Also, in order to improve the task of entity search, we could further extend the model to consider not only the entity but also other neighbor entities. This would mean extending the MF model to consider larger graphs than the current entity's star-graph. Furthermore, we will investigate methodologies for asserting the importance of an attribute automatically.

Acknowledgement

This material is based upon works supported by the European FP7 project LOD2 (257943).

Bibliography

[1] Cafarella, M.J., Halevy, A., Madhavan, J.: Structured Data on the Web. Communications of the ACM **54**(2) (February 2011) 72

[2] Balog, K., Serdyukov, P., de Vries, A.P.: Overview of the TREC 2010 Entity Track. In: Proceedings of the Nineteenth Text REtrieval Conference (TREC 2010), NIST (2011)

[3] Demartini, G., Iofciu, T., De Vries, A.P.: Overview of the INEX 2009 entity ranking track. In Geva, S., Kamps, J., Trotman, A., eds.: Focused Retrieval and Evaluation. Volume 6203 of Lecture Notes in Computer Science. Springer Verlag, Heidelberg (2010) 254–264

[4] Tran, T., Mika, P., Wang, H., Grobelnik, M.: Semsearch'11: the 4th semantic search workshop. In Srinivasan, S., Ramamritham, K., Kumar, A., Ravindra, M.P., Bertino, E., Kumar, R., eds.: Proceedings of the 20th International Conference on World Wide Web, WWW

2011, Hyderabad, India, March 28 - April 1, 2011 (Companion Volume), ACM (2011) 315–316
[5] Blanco, R., Halpin, H., Herzig, D.M., Mika, P., Pound, J., Thompson, H.S., Tran, D.T.: Entity search evaluation over structured web data. In: Proceedings of the 1st International Workshop on Entity-Oriented Search at SIGIR 2011, Beijing, PR China, . (Juli 2011)
[6] Pound, J., Mika, P., Zaragoza, H.: Ad-hoc object retrieval in the web of data. In: Proceedings of the 19th international conference on World Wide Web, New York, New York, USA, ACM Press (2010) 771–780
[7] Zaragoza, H., Craswell, N., Taylor, M.J., Saria, S., Robertson, S.E.: Microsoft Cambridge at TREC 13: Web and Hard Tracks. In: TREC'04. (2004) –1–1
[8] Macdonald, C., Plachouras, V., He, B., Lioma, C., Ounis, I.: University of Glasgow at WebCLEF 2005: Experiments in Per-Field Normalisation and Language Specific Stemming. In: CLEF. (2005) 898–907
[9] Robertson, S., Zaragoza, H.: The Probabilistic Relevance Framework: BM25 and Beyond. Found. Trends Inf. Retr. **3** (April 2009) 333–389
[10] Amati, G., Van Rijsbergen, C.J.: Probabilistic models of information retrieval based on measuring the divergence from randomness. ACM Trans. Inf. Syst. **20**(4) (October 2002) 357–389
[11] Abiteboul, S.: Querying Semi-Structured Data. In: Proceedings of the 6th International Conference on Database Theory. (1997) 1–18
[12] Klyne, G., Carroll, J.J.: Resource Description Framework (RDF): Concepts and Abstract Syntax. Changes **10**(February) (2004) 1–20
[13] Delbru, R., Campinas, S., Tummarello, G.: Searching Web Data: an Entity Retrieval and High-Performance Indexing Model. Web Semantics: Science, Services and Agents on the World Wide Web **10**(0) (2012)
[14] Pérez-Agüera, J.R., Arroyo, J., Greenberg, J., Iglesias, J.P., Fresno, V.: Using BM25F for semantic search. In: Proceedings of the 3rd International Semantic Search Workshop. SEMSEARCH '10, New York, NY, USA, ACM (2010) 2:1–2:8
[15] Blanco, R., Mika, P., Vigna, S.: Effective and Efficient Entity Search in RDF data. In: Proceedings of 10th International Semantic Web Conference, Bonn, Germany, Springer Verlag, Heidelberg (2011)
[16] Harter, S.: A probabilistic approach to automatic keyword indexing. PhD thesis, TheUniversity of Chicago (1974)
[17] Robertson, S.E., van Rijsbergen, C.J., Porter, M.F.: Probabilistic models of indexing and searching. In: Proceedings of the 3rd annual ACM conference on Research and development in information retrieval, Kent, UK, Butterworth & Co. (1981) 35–56
[18] Robertson, S., Zaragoza, H., Taylor, M.: Simple BM25 extension to multiple weighted fields. In: Proceedings of the thirteenth ACM international conference on Information and knowledge management. CIKM '04, New York, NY, USA, ACM (2004) 42–49
[19] Robertson, S.E., Walker, S.: Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In: Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval. SIGIR '94, New York, NY, USA, Springer-Verlag New York, Inc. (1994) 232–241
[20] Hu, X., Eberhart, R.: Solving Constrained Nonlinear Optimization Problems with Particle Swarm Optimization. In: 6th World Multiconference on Systemics, Cybernetics and Informatics (SCI 2002. (2002) 203–206
[21] David J. Sheskin, C., Hall: Handbook of Parametric and Nonparametric Statistical Procedures, Third Edition. CRC (2003)
[22] Büttcher, S., Clarke, C., Cormack, G.V.: Information Retrieval: Implementing and Evaluating Search Engines. The MIT Press (2010)