FuseM: Query-centric Data Fusion on Structured Web Markup

Ran Yu, Ujwal Gadiraju, Besnik Fetahu, Stefan Dietze
L3S Research Center, 30167 Hannover, Germany
Email: {yu, gadiraju, fetahu, dietze}@L3S.de

***Abstract*—Embedded markup based on Microdata, RDFa, and Microformats have become prevalent on the Web and constitute an unprecedented source of data. However, RDF statements extracted from markup are fundamentally different from traditional RDF graphs: entity descriptions are flat, facts are highly redundant, and despite very frequent co-references explicit links are missing. Therefore, carrying out typical entity-centric tasks such as retrieval and summarisation cannot be tackled sufficiently with state-of-the-art methods and require preliminary data fusion. Given the scale and dynamics of Web markup, the applicability of general data fusion approaches is limited. We present a novel query-centric data fusion approach which overcomes such issues through a combination of entity retrieval and fusion techniques geared towards the specific challenges associated with embedded markup. To ensure precise and diverse entity descriptions, we follow a supervised learning approach and train a classifier for data fusion of a pool of candidate facts relevant to a given query and obtained through a preliminary entity retrieval step. We perform a thorough evaluation on a subset of the Web Data Commons dataset and show significant improvement over existing baselines. In addition, an investigation into the coverage and complementarity of facts from the constructed entity descriptions compared to DBpedia, shows potential for aiding tasks such as knowledge base population.**

I. Introduction

Markup annotations embedded in HTML pages have become prevalent on the Web, building on standards such as RDFa[1], Microdata[2] and Microformats[3], and driven by initiatives such as *schema.org*, a joint effort led by Google, Yahoo!, Bing and Yandex. The Web Data Commons [1], a recent initiative investigating a Web crawl of 2.01 billion HTML pages from over 15 million pay-level-domains (PLDs) found that 30% of all pages contain some form of embedded markup already, resulting in a corpus of 20.48 billion RDF quads[4]. The scale of the data suggests potential for a range of tasks, such as entity retrieval, Knowledge Base Augementation (KBA), or entity summarisation.

However, facts extracted from embedded markup have different characteristics when compared to traditional knowledge graphs and Linked Data [2]. Coreferences are very frequent, but are not linked through explicit statements. In contrast to traditional strongly connected RDF graphs, RDF markup statements mostly consist of isolated nodes and small subgraphs. In addition, extracted RDF markup statements are highly redundant and often limited to a small set of highly popular predicates, such as *schema:name*, complemented by a long tail of less frequent statements. Moreover, data extracted from markup contains a wide variety of syntactic and semantic errors [3].

These distinctive characteristics highlight the challenges when aiming to search knowledge sourced from embedded markup. Initial works have applied traditional *entity retrieval* techniques [4] to embedded markup (WDC corpus). However, given the large amount of flat and highly redundant entity descriptions, practical use of search results obtained in that way is limited [2]. Key issues include identity resolution as well as the vast amount of duplicates and near-duplicates. Therefore, the application of *data fusion* techniques is required to obtain a consolidated and correct entity description when answering entity-centric queries. However, given the dynamics of Web markup data, the validity and correctness of a fact is usually of temporal nature.

For these reasons, any data fusion approach would have to consider efficiency aspects in order to enable frequent iterations of the steps required for crawling, extraction and fusion. This suggests a need for focused approaches, which are able to efficiently obtain fused entity descriptions for a given set of entity-seeking queries. In this work, we present a query-centric data fusion approach ($FuseM$) tailored to the specific challenges of markup data, which adopts entity retrieval methods for obtaining a set of candidate entity descriptions and supervised machine learning to enable selection and fusion of valid facts. The main contributions of our work are threefold:

- **FuseM Pipeline for Query-Centric Data Fusion.** We propose a pipeline for query-centric data fusion that is tailored to the specific challenges arising from the characeristics of structured Web markup. To the best of our knowledge, this is the first approach addressing the task of data fusion on Web markup data specifically.
- **Model & Feature Set for Fusion of Markup Data.** We propose a novel data fusion approach consisting of a classification model and diversification step (Section III). We propose and evaluate an original set of features which consider both relevance and correctness of markup facts and use these in a supervised classification model. Experimental results demonstrate high precison (avg. 88.3%) and recall (avg. 94.3%) of our model, outperforming the state-of-art baselines.
- **Investigating the Potential of Web Markup for KBA.** As part of the discussion of our evaluation results in Section V, we investigate the potential of using fused markup data for augmenting existing KBs such as DBpedia. Our results suggest a significant potential for addressing the KBA task through fused markup data.

[1]RDFa W3C recommendation: http://www.w3.org/TR/xhtml-rdfa-primer/
[2]http://www.w3.org/TR/microdata
[3]http://microformats.org
[4]http://www.webdatacommons.org

II. Problem Definition

Our work considers data extracted from a given Web markup corpus, stored in n-quad format, where each entity description corresponds to a set of $\langle s, p, o, u\rangle$ quadruples and s, p, o, u represent subject, predicate, object and the URL of the document from which the triple has been extracted respectively. For a particular real-world entity e, there exist $n \geq 0$ subjects s from the quadruples $\langle s, p, o, u\rangle$ which represent entity descriptions of e. Let us consider $e_s = \langle s, p_i, o_i\rangle$ to be the entity description of e corresponding to subject s. As input for the data fusion task, consider an entity-centric query q, consisting of a keyword representing an entity label, e.g. *Forrest Gump* and the corresponding *schema.org* entity type, for instance, *s:Movie*. We define the data fusion problem given query q and a given Web markup corpus.

Definition 1: Query-centric Data Fusion on Web Markup For an entity-centric query q we aim at constructing a corresponding entity description e_q consisting of a set of correct and diverse facts $f_i \in F'$. Each fact f_i represents a property-value pair $f_i = \langle p_i, o_i\rangle$ describing the entity identified by query q.

We consider a fact suitable for an entity description, if it fulfills the following attributes: 1) **Relevance.** A fact should be *relevant*, i.e. be a statement about the query entity q, 2) **Novelty.** A fact should present *non-duplicate* information in our entity description, 3) **Correctness.** A fact should be *correct*.

III. FuseM Query-centric Data Fusion

To address the *query-centric data fusion* problem defined above, we propose two main steps, namely (i) entity retrieval, and (ii) data fusion to construct a fused entity description e_q.

A. Entity Retrieval

The first step is a prerequisite for an efficient query-centric data fusion process by providing a pool of candidate entity descriptions $e_s \in E$ and consequently facts. We exploit a standard IR index over type-specific subsets of a given Web markup corpus through Lucene. We query our index for q using the state-of-the-art BM25 model on specific fields in the entity descriptions, for instance, *s:name* or *s:alternateName*. As such fields vary between types, we have provided a full list online at http://l3s.de/~yu/fuseM/. The entity descriptions $e_s \in E$ retrieved in this step are associated with a set of facts F, where $f_i = \langle p_i, o_i\rangle$ constitutes a candidate fact for the description of q. A necessary step to further improve the retrieved entity descriptions is to resolve the object properties. Earlier work [3] has shown that object properties are frequently misused as datatype properties referring to literals rather than nodes. For instance, o_1 in $\langle s_1, s : actor, o_1\rangle$ might refer to either a node or a literal. To homogenize data for further processing, we obtain the label $(s : name)$ for any nodes which are referred in candidate entity description and replace corresponding node references with the obtained literal.

B. Data Fusion

In this step, we seek to select the complete set of distinct correct facts F' from the candidate set F, taking into account issues such as *redundancy, coreferences, lack of links and errors*. The fusion process consists of a classification and a diversification step which are described below.

1) Classification: To address the aforementioned issues, we learn a supervised model that produces a binary classification for a given fact $f \in F$ into one of the labels {*'correct'*, *'incorrect'*}. Table I shows the list of computed features, where the identified feature categories directly correspond to the attributes described in Section II.

TABLE I: Data fusion features.

Category	Notation	Feature description
Relevance	t_1^r	BM25 score based on the entity retrieval result
	t_2^r	Rank of highest ranked entity that contains fact f based on the entity retrieval result
	t_3^r	Similarity $Sim(e_s)$ between the retrieved entity e_s and the corresponding DBpedia page of query entity q. $Sim(e_s)$ is computed as in Equation 1
Clustering	t_1^c	Normalized cluster size that f belongs to
	t_2^c	Number of clusters in the cluster result of predicate p
	t_3^c	Average cluster size
	t_4^c	Variance of the cluster sizes
Quality	t_1^q	Maximum PageRank score of the PLDs containing fact f
	t_2^q	Maximum size (number of facts) of e_s containing f
	t_3^q	Predicate frequency in F
	t_4^q	Fact frequency in F

Relevance Features. Here we consider three features that encode the relevance of a fact for q. As the first feature in this group, we use the BM25 score computed between q and e_s from which we extract f. Consequently, we extract the largest BM25 score and the highest rank of an entity e_s containing fact f as feature, i.e. t_1^r and t_2^r. For query entities which have a representation in DBpedia, we measure the overall similarity between e_s and the representation of q in a given KB (DBpedia), i.e. feature t_3^r. The score corresponds to the ESA measure [5] which is computed as in Equation 1.

$$Sim(e_s) = \sum_{i,j} Sim(f_i, f_j) \tag{1}$$

where $f_i \in q$, $f_j \in e_s$, and $Sim(f_i, f_j)$ is computed as:

$$Sim(f_i, f_j) = \begin{cases} w(p_i) \cdot cos(o_i, o_j) & \text{if } p_i = p_j \\ 0 & \text{if } p_i \neq p_j \end{cases} \tag{2}$$

Here, $w(p_i)$ is the weight of predicate p_i, $cos(o_i, o_j)$ is the cosine similarity between object value o_i and o_j. We assign the weight $w(p_i)$ manually according to the importance of predicate p_i on measuring the relatedness between entity descriptions.

Clustering Features. Given the diverse and heterogeneous nature of Web markup, our candidate set contains vast amounts of near-duplicate facts, often using varied surface terms for the same or overlapping meanings. For instance, *Tom Hanks* and *T. Hanks* are equivalent surface forms representing the same entity. We approach this problem through clustering, to group or canonicalize different literals or surface forms for specific object values. To detect duplicates and near duplicates, we first cluster facts that have the same predicate p into n clusters $(c_1, c_2, \cdots, c_n) \in C$. Another challenge considered here is the cardinality of predicates. Depending on the predicate, the number of potentially correct statements varies. For example, *s:actor* is associated with multiple values, whereas *s:duration*

normally has only one valid statement. We employ the X-Means algorithm [6], that is able to automatically determine the number of clusters. Based on the clustering result, we extracted features t_i^c. We consider the size of a cluster as feature t_1^c indicating the frequency of a fact, as well as fact distribution features $t_i^c, i = 2, 3, 4$ as quality indicators of a predicate.

Quality Features. We consider the PageRank score (feature t_1^q) as an authority indicator of the PLD from which a fact is extracted. Furthermore, we analyze the frequency of the predicates and the fact values. This is based on the assumption that correct facts are likely to be more frequent, hence we introduce frequency-based features $t_i^q, i = 2, 3, 4$.

From the computed features we train a supervised classifier for classifying the facts from F into the binary labels {*'correct'*, *'incorrect'*}. We experimented with several state-of-the-art classification algorithms (SVM, kNN with varying ks and Naive Bayes). Since SVM achieves a precision score that is 3% higher than Naive Bayes, and 14% higher than the best KNN($k = 3$), we rely on a trained SVM classifier which uses a linear kernal function.

2) Diversification: The diversification step ($FuseM_{DIV}$) aims at reducing the amount of duplicates and near-duplicates after applying the classification step ($FuseM_{SVM}$), in order to improve the diversity of our resulting entity descriptions. Here, we take into account features obtained from the *clustering feature set* described above. Our approach consists of selecting exactly one correct fact from each cluster. To avoid near duplicates, as part of this diversification step ($FuseM_{DIV}$) we select the fact closest to each cluster center to build the final entity description e_q for q.

IV. Experimental Setup and Evaluation

We evaluate the performance of $FuseM$ with respect to accuracy and diversity of the fused entity descriptions.

A. Dataset & Queries.

Dataset. Our experiment is built upon the WDC2014 dataset. Specifically, we extracted 3 type specific subsets that consist of entity facts about instances of the *schema.org* types *s:Movie* (3,540,733 subjects, 76,586,127 quads), *s:Book* (7,411,863 subjects, 68,816,749 quads) and *s:Product* (287,815,069 subjects, 2,829,523,589 quads) respectively. We choose the types *s:Movie, s:Book* since initial experiments indicated that these types are well-reflected in the WDC2014 datasets, and at the same time, their facts are comparably easy to validate manually when attempting to label a ground truth. To evaluate performance on data which typically is not well represented in traditional KBs, we also consider a subset of instances of type *s:Product*.

Query Set. To evaluate the performance and coverage gain, we use 3 groups of queries for the experiment.

- **Book**, **Movie**. These two query sets represent randomly selected entities of type *s:Book* and *s:Movie* from Wikipedia. To construct the actual query from each entity, we use the entity label, e.g. "Man of Steel", as query term and the entity type, mapped to its schema.org type as query type restriction.
- **Product**. We randomly select 30 names of products under the requirement that each appears in at least 20 different PLDs in WDC, to ensure that there is sufficient consensus on the name being a legitimate product title. Manual inspection confirmed that none of such products is represented in Wikipedia/DBpedia, thus the feature t_3^r in Table I is not included in experiments conducted on the *Product* dataset.

B. Approach Configuration & Baselines

Approach Configurations. We evaluate the precision at each step, namely the classification ($FuseM_{SVM}$) and the final entity description after diversification ($FuseM_{DIV}$). In order to evaluate the performance of features, we also run the approach under different combinations of feature categories as listed in Section III-B1.

Baselines. We consider distinct facts obtained through 2 different baselines. Note that given the novelty of the task and data, the state-of-the-art and available baselines are strongly limited. To the best of our knowledge, the $CBFS$ approach [7] is the only available method so far directly geared towards the task presented in this paper.

- CBFS@k: facts selected based on the $CBFS$ approach [7] with candidates from the top k retrieved entity descriptions. The $CBFS$ approach first clusters the associated values at the predicate level into n clusters $(c_1, c_2, \cdots, c_n) \in C$. Then the facts that are closest to the cluster's centroid from each cluster that meet the following criteria are selected:

$$|c_j| > \beta \cdot \max(|c_k|), c_k \in C \tag{3}$$

where $|c_j|$ denotes the size of cluster c_j, and β is a parameter used to adjust the number of facts. In our experiments, β is empirically set at 0.5.
- BM25@k: distinct facts from the top k entity descriptions according to the BM25 retrieval results.

C. Ground Truth & Metrics

Ground Truth. We built a ground truth by acquiring labels for all distinct facts from the retrieved candidate set F, as either *correct* or *incorrect* with respect to a given query. Three authors of this paper acted as experts and designed a coding frame according to which we could decide whether or not a fact was correct with respect to q. After resolving disagreements on the coding frame and a subset of each dataset, every fact was associated with one expert label through manual deliberation. We followed the guidelines laid out by Strauss [8] during the coding process. Distinct facts were obtained by removing duplicate literals, null values, URLs and the unresolved objects. All query sets and the ground truth are available online at http://l3s.de/~yu/fuseM/. For the diversity evaluation, we followed a similar coding process to label the $f_i \in F'$ of different methods and identify the duplicate facts. To limit the manual labor without compromising on the size of the data, we randomly selected 10 queries from each query set and labeled facts of the corresponding descriptions obtained by each method.

Metrics. We apply *10-fold cross validation* for different approaches and use standard precision P - the percentage of $f_i \in F'$ that correctlly describe q. Furthermore, we measure the diversity ($Dist\%$) of e_q as the percentage of *distinct correct facts* among the correct facts in F'.

D. Results

The precision results of different approaches are shown in Table II. $FuseM_{SVM}$ shows significant improvement on precision compared to the baselines, i.e. achieves a gain of 44.2% over $BM25$, and 42.8% over $CBFS$ on average @20. The gain is larger in the presence of more noise in the candidate set, as can be observed based on the larger precision gain of $FuseM_{SVM}$ over baselines @50, i.e. 59.1% over $BM25$ and 49.2% over $CBFS$.

TABLE II: Precision of query-centric data fusion approaches.

Type	BM25@20	CBFS@20	FuseM$_{SVM}$@20	FuseM$_{DIV}$@20
Product	0.944	0.937	**0.991**	0.99
Book	0.179	0.181	**0.755**	0.729
Movie	0.202	0.251	**0.906**	0.904
Type	**BM25@50**	**CBFS@50**	**FuseM$_{SVM}$@50**	**FuseM$_{DIV}$@50**
Product	0.628	0.876	**0.983**	0.979
Book	0.094	0.121	0.784	**0.787**
Movie	0.107	0.129	0.835	**0.863**

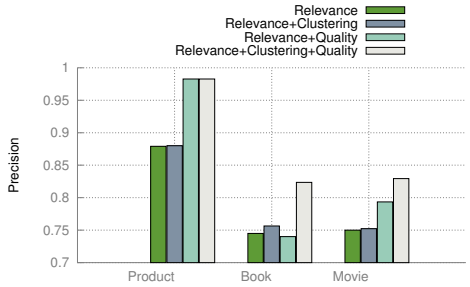

Fig. 1: Precision of different feature combinations for $FuseM_{SVM}$.

Furthermore, we also investigate the performance of different feature configurations. Since relevance is a key requirement, relevance features are included in all the configurations. Figure 1 shows that the precision improves through each additional feature category. For all the datasets, the highest precision is acquired when using features from all categories.

TABLE III: Diversity of obtained entity descriptions ($Dist\%$).

Type	BM25@50	CBFS@50	FuseM$_{DIV}$@50
Product	83.3	**88.2**	**88.2**
Book	78.5	90.1	**92.5**
Movie	81.6	91.1	**96.4**

Finally, Table III shows the diversity evaluation results. Our $FuseM_{DIV}$ approach shows an 11.3% improvement compared to $BM25$ and 2.4% compared to $CBFS$. This increase in diversity suggests that, the clustering based approach contributes to improving the diversity of the entity descriptions. This effect is even more evident on very popular entities, that correspond to a large amount of coreferences and hence, entail near-duplicate facts.

V. Coverage Gain

We use *coverage gain (CG)* as the percentage of detected facts that are not available in DBpedia. For 10 queries from each query set, we manually compared the data fusion results ($FuseM_{DIV}$) with corresponding DBpedia resources to determine the CG. Precisely, we labeled facts generated by ($FuseM_{DIV}$) into three categories: *existing*, i.e. facts already represented in DBpedia, *new*, i.e. new facts for existing DBpedia properties (e.g. an ISBN number), and *new-p*, i.e. new facts involving predicates not yet in use in DBpedia. The distribution of facts for three types under consideration is presented in Figure 2.

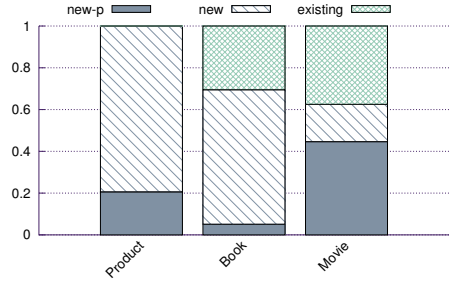

Fig. 2: Distribution of facts constituting entity descriptions produced with $FuseM$, compared to facts existing in DBpedia.

We found that, on average, 66.1% of the facts detected by our approach for queries that have corresponding DBpedia resources do not exist in DBpedia. It is noteworthy however that, among the new predicates, we observe a large proportion of dynamic and Web-specific properties, such as user-generated reviews, which might seem of less value for particular KBA tasks. We also calculate the extra coverage considering the predicates that exist in DBpedia for the given query type, leading to a CG of 41.7% on average. This suggests that markup data provides diverse information able to complement the knowledge available in DBpedia and potentially, other KBs.

VI. Conclusions

In this work, we have introduced $FuseM$, an approach towards query-centric fusion of data from Web markup with the overall aim of providing rich, correct and diverse entity descriptions. Results on the WDC2014 corpus suggest superior performance of our approach with respect to diversity as well as precision compared to state-of-the-art baselines. In addition, our initial assessment of the coverage gain suggests potential to aid KBA tasks through Web markup in general and our approach in particular.

References

[1] R. Meusel, P. Petrovski, and C. Bizer, "The webdatacommons microdata, rdfa and microformat dataset series," in *ISWC*, 2014.

[2] R. Yu, B. Fetahu, U. Gadiraju, and S. Dietze, "A survey on challenges in web markup data for entity retrieval," in *ISWC Posters & Demonstrations Track*, 2016.

[3] R. Meusel and H. Paulheim, "Heuristics for fixing common errors in deployed schema.org microdata." in *ESWC*, 2015.

[4] R. Blanco, P. Mika, and S. Vigna, "Effective and efficient entity search in rdf data," in *ISWC*, 2011.

[5] E. Gabrilovich and S. Markovitch, "Computing semantic relatedness using wikipedia-based explicit semantic analysis," in *IJCAI*, 2007.

[6] D. Pelleg, A. W. Moore *et al.*, "X-means: Extending k-means with efficient estimation of the number of clusters." in *ICML*, 2000.

[7] R. Yu, U. Gadiraju, X. Zhu, B. Fetahu, and S. Dietze, "Entity summarisation on structured web markup," in *ESWC Poster Track*, 2016.

[8] A. L. Strauss, *Qualitative analysis for social scientists*. Cambridge University Press, 1987.