The Extremes of Good and Evil

Master Thesis

presented by
Oliver Frendo (1510432),
Dandan Li (),
Zehui Wang () *and*
Yi-Ru Cheng ()

submitted to the
Data and Web Science Group
Prof. Dr. Right Name Here
University of Mannheim

November 27, 2015

Contents

List of Figures

List of Tables

Chapter 1

Web Data Integration

1.1 Introduction to data and use case

1.2 Data translation

1.2.1 Data collection

In order to collect suitable data we tried different data service providers such as Datahub, finally we have collected total four datasets from three different sources and in three different formats, including:

Forbes: Company

The Forbes offers a .xls file with a list of Top 2000 companies during the period 2000 to 2014 which were published in Forbes magazine because of great performance in terms of business achievements. This dataset describes the basic information about these top 2000 companies. For example, location shows where this company is founded, industry depicts what fields the company focus on and so on.

DBpedia: Company

The information of company is extracted from DBpedia, since it provides relatively complete information. To access information from DBPedia we used the public SPARQL endpoint (at http://dbpedia.org/sparql). Figure 1.1 is our query for company, actually there is total 764398 companies in DBpedia, which would be too much for us and also not easy to handle it in terms of processing time and space. In order to reduce the number of data, we limit the company types to ”company” and ”public company” and only extract the companies that provide attributes ”LocationCity” and ”LocationCountry”, these two attributes can also be related with Location Information, that's why we consider them as necessary and others are optional. On the other hand, if all these attributes are necessary, there will be only few thousands companies extracted, because not all companies have all these nine attributes, in this case few overlapping data will be in the final integration results. In addition to this, as many attributes such as KeyPeople, locationCity have multiple values which result in the same company would appear more than one times, to avoid these duplicates we used ”group_contac”, a function in Sparql, to group many value together. There are also many values for Revenue but without date notation, so we just took the maximum value.

Freebase: Company

Freebase, like DBPedia, offers a web service which can be queried for data. Instead of SPARQL, however, the service is used by sending JSON requests. Like DBPedia this lets us select certain companies, or companies with certain attributes.

DBpedia: Location

We also extracted Location information from DBpedia with the same method as Company. Figure 1.2 is the query for location. For the same reason as Company, we limit the location types to ”city” and ”AdministrativeRegion”, which are more relevant to our company dataset. Also some attributes have many values without extra information, it's hard to identify which one represents the current state, thus, we just took the maximum number of them among multiple values. Furthermore, the name of locations are provide in different languages, while in our project we just focus on english, so we filtered language as english.

Table 1.1: Basic Profile of Each Dataset

	Source	Format	Class	#Entities	#Attributes
	List of Attributes				
Forbes Global	forbes.com	xlsx	company	2000	7
	Company,country,industry,revenue,assets,marketvalue,profit				
DBpedia	dbpedia.org/sparql	csv	company	16051	9
	Company,country,industry,revenue,numberOfemployees, foundingYear,assets,keyPeople,location				
Freebase	freebase.com/query	json	company		9
	Company,country,industry,revenue,numberOfemployees, foundingYear,profit,keyPeople,location				
Dbpeida	dbpedia.org/sparql	csv	location	3270	5
	Location,country,population,area,elevation				

1.2.2 Integrated schema

We looked into four datasets and did the following Integrated Schema.In this table we use prefix dataset 1, 2, 3,4 respectively represent Forbes, DBpedia(company), Freebase and DBpedia(Location)

1.2.3 Data transformations

In this phase, two kinds of transformations occurred. One is processed during mapping, another one is processed with Java Programming. First of all, during mapping process, since revenue and assets are written in scientific notation, while in our target schema it should be in original data, therefore we used some functions in Mapforce to convert scientific notation into decimals. Secondly, the original data has no ID, but it will be used later, so we also append ”ID” for these four dataset individually with ”GenerateID” function provided by MapForce. In addition, many

attributes in Dbpedia are in the form of URL, while only the values are meaningful, thus, we removed the URL part with Java and extracted values we want, in the mean time, we also replaced punctuation and symbols like "_" or ".". Furthermore, some words, that have various spellings such as US, USA, U.S, America, were replaced with unified United States of America.

```
SELECT   ?company
group_concat(distinct str(?locationCity);separator=";;") as ?locationCity
group_concat(distinct str(?locationCountry);separator=";;")
as ?locationCountry
group_concat(distinct str(?keyPeople);separator=";;") as ?keyPeople
group_concat(distinct str(?foundingYear);separator=";;")
as ?foundingYear
group_concat(distinct str(?industry);separator=";;") as ?industry
max(?revenue) as ?revenue     ?numberOfEmployees    ?asset
WHERE {
{?company rdf:type dbo:Company}
UNION
{?company dbr:type dbr:Public_company} .
optional{?company dbo:numberOfEmployees ?numberOfEmployees .}
?company dbp:locationCity ?locationCity.
?company dbp:locationCountry ?locationCountry.
optional{?company dbp:keyPeople ?keyPeople .}
optional{?company dbo:foundingYear ?foundingYear .}
optional{?company dbo:industry ?industry .}
optional{?company dbo:revenue ?revenue .}
optional{?company dbo:assets ?asset .}
}ORDER BY ?company
```

Figure 1.1: DBpedia Query For Company

```
SELECT distinct ?label Max(?pop)as ?population
Max(?areaTotal)as ?areaTotal
Max(?elevation)as ?elevation ?country
WHERE {
{?location rdf:type dbo:City .}
union
{?location rdf:type dbo:AdministrativeRegion .}
?location rdfs:label ?label .
?location dbo:areaTotal ?areaTotal .
?location dbo:elevation ?elevation .
?location dbo:country ?country .
?location dbo:populationTotal ?pop .
FILTER((lang(?label) = 'en'))
}
```

Figure 1.2: DBpedia Query For Location

Table 1.2: Integrated Schema

Class Name	Attributes Name	Datasets in which attribute is found	Conflict resolution strategies
company	(company)name	dataset 1, 2, 3	LongestString, FavourSources, Voting
company,location	country	dataset 1, 2, 3, 4	Voting
company	industries	dataset 1, 2, 3	Union
company	Sales/ revenue	dataset 1, 2, 3	FavourSources
company	number Of Employees	dataset 2, 3	Average, Growth: Max
company	founding year/ date founded	dataset 2, 3	MostComplete (complete date) AND, MostComplete (sample)
company	Assets	dataset 1, 2	FavourSources
company	Market Value	dataset 1	/
company	profit	dataset 1, 3	FavourSources
company	continent	dataset 1	/
company	keyPeople/ leadership	dataset 2, 3	Union
company, location	name	dataset 2, 3, 4	Union AND FavourSources
location	population total	dataset 4	/
location	area total	dataset 4	/
location	elevation	dataset 4	/

1.3 Identity resolution

1.3.1 Gold standards

Forbes ↔ Freebase

Freebase ↔ DBpedia companies

DBpedia companies ↔ DBpedia locations

1.3.2 Matching rules

Aim here was to generate correspondences between companies of two different sources.

Comments: COUNTRIES May have an effect on companies that have multiple countries and thus the number of correspondencies INDUSTRIES Needed a better measure than just Jaccard or Levenshtein because of misspellings or slight differences, e.g. ”Transport” and ”Transportation” REVENUE/PROFIT Numeric data from Freebase is too sparse, unreliabe or outdated. Learning a matching rule in RapidMiner confirms this by assigning weights of 0 to both these attributes.

LOCATIONS: No need for Levenshtein NAME: New York and New York City need Jaccard. Probably no misspellings though

HARD CASES Central Japan Railway, EAST JAPAN RAILWAY (FALSE) Vodafone, Vodafone Group plc (TRUE) Syracuse, Utah AND Syracuse, New York ”Chicago and Nashville” as single attribute from DBpedia

EXAMPLE FOR HIGHEST JACCARD C1=New York, Chicago. C2 New York. Should be correspondence, so similarity = 1 Couldn’t just take highest equals because of New York and New York City for example

COMBINATION OF LEVENSHTEIN AND JACCARD: Why? Why are they better than normal values?

1.3.3 Blocking functions

Notes on blocking function:

Freebase vs Forbes: - Country is always available in Forbes, sparse in Freebase (EXACT NUMBER)

Freebase vs DBPedia: - Country is somewhat sparse in both datasets (EXACT NUMBER) - DateFounded somewhat sparse in both (EXACT NUMBER)

Company vs Location - Country is only way to go

Own idea: Implement own partitioning blocker, but more fuzzy logic - I.E. Country first, then date, then location or so as alternative (EXACT NUMBER) -

Attribute	MatchingRule	P	R	F1
Forbes vs Freebase				
name	Equals	1,0000	0,7500	0,8571
	Levenshtein	0,8571	1,0000	0,9231
countries	Equals	0,8571	1,0000	0,9231
	Jaccard	0,8571	1,0000	0,9231
	Highest Jaccard	0,8571	1,0000	0,9231
industries	Jaccard	0,9091	0,8333	0,8696
	Combination of Jaccard and Levenshtein	0,8571	1,0000	0,9231
revenue/ profit	PercentageSimilarity (max_percentage=0.5)	0,8571	1,0000	0,9231
Freebase vs DBpedia				
revenue/ numberOfEmployees	PercentageSimilarity (max_percentage=0.5)	0,9167	0,9167	0,9167
dateFounded	YearSimilarity (maxDifference=20)	0,9167	0,9167	0,9167
keyPeople	Jaccard	0,9167	0,9167	0,9167
	Combination of Jaccard and Levenshtein	0,9167	0,9167	0,9167
locations	Jaccard	0,9167	0,9167	0,9167
	Highest Jaccard	0,9167	0,9167	0,9167
DBpedia companies vs DBpedia locations				
countries	Highest Jaccard	0,9706	0,9429	0,9565
locations	Jaccard	0,9630	0,7429	0,8387
	Highest Jaccard	0,9706	0,9429	0,9429

Table 1.3: Matching rule accuracies

Generate more pairs, matching rule will take care if its same company or not- more conservative approach

1.3.4 Learning matching rules

1.4 Data fusion

1.4.1 Input data

1.4.2 Gold standard

1.4.3 Conflict resolution functions

1.4.4 Accuracy

Bibliography

Ehrenwörtliche Erklärung

Ich versichere, dass ich die beiliegende Master-/Bachelorarbeit ohne Hilfe Dritter und ohne Benutzung anderer als der angegebenen Quellen und Hilfsmittel angefertigt und die den benutzten Quellen wörtlich oder inhaltlich entnommenen Stellen als solche kenntlich gemacht habe. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen. Ich bin mir bewusst, dass eine falsche Er- klärung rechtliche Folgen haben wird.

Mannheim, November 27, 2015 Unterschrift