Using the Structure of DBpedia for Exploratory Search

Samantha Lam
DERI, NUI Galway
IDA Business Park, Lower Dangan
Galway, Ireland
samantha.lam@deri.org

Conor Hayes
DERI, NUI Galway
IDA Business Park, Lower Dangan
Galway, Ireland
conor.hayes@deri.org

ABSTRACT

Recently there has been much work in defining similarity on heterogeneous networks in a supervised manner, using prescribed patterns based on the network schema. However, there is also a wealth of data that doesn't ascribe rigidly to fixed schemas, such as DBpedia. In this work we explore the idea of using DBpedia for computing semantic relatedness in the context of an exploratory search. We first analyse the main datasets of DBpedia, as graphs, to assess their suitability for the task. We then employ a weighting scheme on the Infobox properties graph inspired by the text retrieval intuition that rare terms, or in our case edges, are more important for retrieving related items. Lastly, we cluster the related nodes and show that we can further enhance the meaning of the clusters to the user by using the Wikipedia Categories and DBpedia Ontology to label them.

1. INTRODUCTION

DBpedia is a project that aims to to extract structured information from the web's largest online encyclopedia, Wikipedia. With a rich RDF description of the entities and classifications in concept hierarchies found in its datasets, there is much potential for information extraction. More importantly, this means that Wikipedia can be represented as a multigraph, i.e. a heterogeneous network where the nodes and edges can represent more than one type (also known as a typed graph). Past methods of knowledge extraction from DBpedia have made use of this representation for information retrieval in areas such as movie recommendation [10, 13], topic labelling [5] and exploratory search [4, 8, 2].

However, many of these past methods present in the semantic similarity and exploratory search domain don't fully exploit the typed nature of the graph and treat it as an untyped (homogeneous) one [8]. In particular DBpedia is a semantic graph and as such the links can be used to infer similarity between entities, e.g. for musicians in DBpedia, there is a link type called *AssociatedActs*, as well as *RecordLabel*. These links show that there are relationships between the musician and another musician, and between the musician and its record label, but these are clearly two different relationships. Past methods also tended to focus solely on one ontology e.g. the Wikipedia Category [22], or when utilising the Resource Description Framework (RDF), not use a graph-based model [15, 21]. These methods are useful for queries that have a specific answer, e.g. logical queries for finding answers to questions such as 'what year was X person born in?'. However, these methods are unsuitable for more general queries such as 'what kind of artists are similar to Lisa Hannigan', where given the large availability of similarity measures in ranking, a graph-based approach may be more beneficial.

Recently, efforts in addressing the problem of measuring similarity on a heterogeneous graph typically relies on a fixed schema with known path patterns of interest. However, given a heterogeneous network with no clearly defined schema, such as DBpedia, how do we find similar nodes? Thus, the key difference between the method we propose and these past works is that ours is an unsupervised one. Supervised methods are useful for when the the user is familiar with the query entity and its relations, but for when the user doesn't know the scope of the query's domain, we can utilise an unsupervised method in the following way:

- find similar objects to a given query
- classify these related objects into labelled clusters resulting in an enhanced navigational search.

This is the type of exploratory search scenario we aim to address. Hearst [3] give a concise overview of the comparison between clustering and faceted categories in the context of information exploration and notes that unexpected trends in data may not be found due to the fact that faceted categories are known in advance. Thus, we take a clustering approach to the exploratory search, because as far as we know, there has been little work done in the explicit clustering using the types/edge labels as features.

In short, we analyse the underlying graph structure of DBpedia according to its different ontologies to assess its suitability for similarity computation. We exploit the heterogeneity of the frequency distribution of the link types of the Infobox

To appear in MDS Workshop in conjunction with SIGKDD 2013

properties dataset for ranking and clustering. Then, we use the DBpedia Ontology and Wikipedia Categories to label these clusters. The context of this work is for an exploratory search and we present some initial results that shows that it can be a competitive alternative to previous faceted search approaches.

2. MOTIVATION/CONTRIBUTION

A distinct feature of the DBpedia graph is that despite the existance of schemas, due to its crowd-sourced creation, the schemas are not adhered to strictly enough to apply previous methods without major selective preprocessing of the graph. We are primarily interested in a ranking and clustering method that is general as possible due to the large scope of exploratory search. In order to do this, we need to understand how the different datasets/ontologies of DBpedia relate to each other, as networks. From our analysis, we further investigate the Infobox dataset as a means of computing semantic similarity for ranking in the context of an exploratory search.

Thus, we need to first study the following questions:

- What is DBpedia's structure with respect to its different schemas? E.g. how do the network properties of the YAGO and DBpedia ontologies differ from the Infobox network?
- Can we provide an effective similarity ranking process in light of these findings? I.e., can we use these findings to aid exploratory search on Wikipedia?

In short, our contribution is a general framework for exploratory search on Wikipedia using DBpedia consisting of:

- A ranking process that exploits the non-uniform frequency distribution of the link types over the Infobox dataset.
- A method to cluster these rankings using the link types as an analog to term frequency in documents, and then mapping members of the clusters back to the Wikipedia Categories as a means of labelling.

3. RELATED WORK

We present work related to the problem of using DBpedia for exploratory search on Wikipedia, roughly corresponding to two separate research areas.

3.1 Similarity and Ranking

Ramakrishnan et al [14] take a similar approach to our ranking problem in that both our works use the intuition that rarer types are more informative for similarity calculation. The specific problem they address is given two query nodes, to retrieve a subgraph that best summarises their relationships. Our problem is different as we are given one query node, and we summarise *its* relationships in the contexts of the nodes found to be similar to it. Leal et al [6] define a proximity measure based on the number of paths that exist between two nodes to compute relatedness. They show the effectiveness of their measure in the music domain but notes that manual configuration is needed in order to apply their method for different domains, i.e. the relevant edge types need to be specified by the user.

Recent work on heterogeneous network similarity search such as PathSim [16] are also of interest. PathSim is a method for computing similarity on networks that have a known schema and uses the notion of meta-paths which specifies patterns of paths as a contribution to the similarity. Similarly, Minkov and Cohen [9] used a supervised lazy graph walk approach on heterogeneous graphs to rank items relevant to specific search queries such as finding email aliases in a given email network schema. Another recent approach PathRank [7] also requires a similar 'path-guide' for filtering out irrelevant paths that don't contribute to the similarity.

Apart from ranking methods, the work whose motivation is conceptually most similar to ours would be RankClus [17] and NetClus [18] which requires specific bi-typed and star network schema, respectively.

As mentioned in the introduction, these past efforts in addressing the similarity ranking problem on a heterogeneous network relies on a fixed schema and known path patterns of interest, of which is difficult to define when the schemas on DBpedia are unclear, particularly in the context of an exploratory search. We focus on how to address DBpedia's lack of a fixed schema by exploiting the inhomogeneous frequency distribution of the link types.

3.2 Exploratory Search on Wikipedia

Faceted search in which the facets or categories are generally known in advance, e.g. in filtering sections for shopping websites, has previously been applied on Wikipedia as a form of exploratory search. Facetedpedia [8, 21] is a good example of such an approach. The authors use the Wikipedia pagelinks and Categories for their framework such that for each article, the facets are created from the categories. Similarly, [2] presented an interface called Faceted Wikipedia Search[1] which, although not explicitly noted in the paper, has facets that are created from the names of the link types. The main difference between their work and ours is that the facets are created as the user interacts with the search whereas we create and label clusters as a comparison to facets.

The semantic similarity between two nodes has also been used for exploratory search. Mirizzi et al. [11] uses a combination of external sources and Wikipedia pagelinks for matching according to the *rdfs:label* link of two nodes to compare their similarity. Heim et al [4] uses a path-based approach to show relationships between two entities. Schonhofen [15] again uses the Wikipedia Category network to identify document topics based on the term frequency between Wikipedia articles and Categories.

A common theme in these past methods is that they do not take heterogeneity of link types into account, are applied on full Wikipedia graph, or are text-based. Moreover, a common requirement for the faceted search approach is the pre-specification of facets, which is counter-productive to an unsupervised approach.

[1] Although it has since been removed as of 2012

4. FRAMEWORK

Our general framework in the exploratory search consists of:

1. An input query q which is a Wikipedia article, and number of clusters k.
2. A ranking mechanism that retrieves similar items on the Infobox properties graph.
3. Items retrieved are clustered using the edge types of their immediate neighbours as features.
4. Labels for each cluster are generated by mapping the edge types to the Wikipedia Categories and DBpedia Ontology graph.

We employ the $k-$step markov centrality for the initial ranking [20] because we believe that finding nodes that have high *relative* (to the query node) centrality is a good indicator of similarity. E.g. Lisa Hannigan may have immediate neighbour nodes consisting of her musical genre and associated artists, but taking $k = 2$ steps out, if a lot of her associated artists also have high overlapping musical genres, then we can deduce that that musical genre is related to Lisa's style of music.

Then in order to quantify the weights, we use the analog of types and ontology to term and documents in the TF-IDF weighting scheme commonly used in text mining. The term frequency $tf(t, d)$ = frequency of term t in document d, is the number of occurrence of types in our case. We describe this process in detail in Section 7.

5. DATA

DBpedia is a crowd-sourced project that extracts information from Wikipedia[2] as a structured knowledge base. Each 'thing' in the DBpedia data set is denoted by a de-referenceable URI-based reference of the form `http://dbpedia.org/resource/Name`, where `Name` is derived from the URL of the source Wikipedia article, and has the form `http://en.wikipedia.org/wiki/Name`. Thus, each DBpedia entity is tied directly to a Wikipedia article.[3]

The version of DBpedia we use is the most recent available.[4] We first briefly describe the RDF model and then how DBpedia is organised and how they are represented as a typed graph in its different vocabularies.

RDF and RDFS

The Resource Description Framework (RDF) is a general metadata data model for describing information on the web. It is based upon making statements about the resource in question in an object-orientated manner, i.e. entity-attribute-value or object-predicate-subject. The subject of an RDF statement is either a URI or a blank node[5], which denote resources. The predicate is a URI which also indicates a resource, representing a relationship. The object is a URI, blank node or a Unicode string literal. Figure 1 summarises an RDF statement in its graphical representation. Thus a collection of such statements results in a directed, multigraph. This framework is the most basic conceptual model and the RDF Schema (RDFS) extends it, which in turn is used as a basis for RDF vocabularies for describing ontologies.

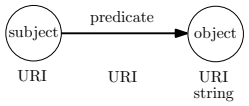


Figure 1: Diagram of an RDF entity relationship

DBpedia provides different classification schemata for things due to the different extraction and application requirements (see [1] for a detailed description). For example, the Wikipedia Categories arose from a collaborative effort from the editors of Wikipedia, YAGO was a specifically designed ontology for linking Wikipedia with other ontologies, and the DBpedia Ontology was manually created from the again crowd-sourced emergent Infobox.

The largest consistent overlap between the different graphs are the Wikipedia articles (there are certainly mappings from YAGO to the DBpedia Categories, but we decided to keep the two separate for consistency). The following is a summary of the datasets and their extraction:

- **DBpedia (Infobox) Ontology**: An Infobox in Wikipedia is a box that summarises the article and can aid navigation to other interrelated articles.[6] The ontology was manually created based on the most commonly used Infoboxes within Wikipedia.
- **Wikipedia Categories**: Maintained by Wikipedia editors, does not form a proper topical hierarchy (there are cycles).
- **YAGO**: An ontology derived from Wikipedia and WordNet. It forms a deep subsumption hierarchy.

Extraction of DBpedia networks

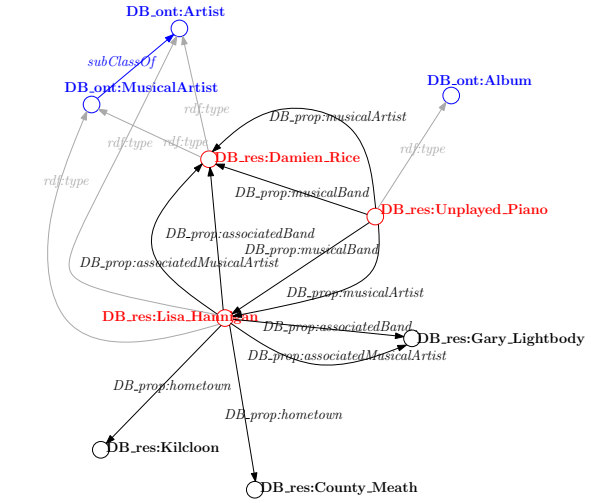
Our datasets were obtained from the DBpedia downloads page[7] and the datasets we explicitly use are the following English versions: DBpedia Infobox Ontology, Wikipedia Categories and YAGO. The properties of the node and edge types of each of the datasets is summarised in Table 2. It is noted that Categories and YAGO structures have been significantly reduced due to the omission of edges that lead to nodes that are not related to the Wikipedia graph (label, prefLabel, sameAs) and in order to avoid large sink/sources in the graph (*rdfs:type* which point to the one node *Concepts*). We also omit any resources that are strings for the same reason.

6. DBPEDIA NETWORK SELECTION

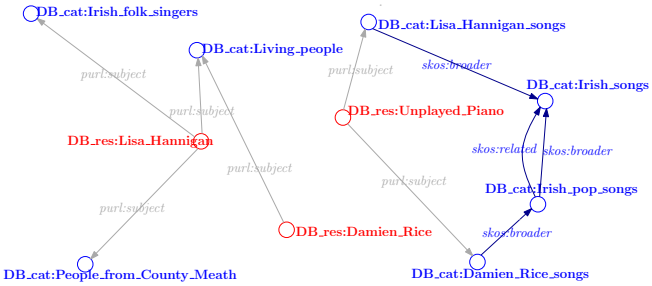
We now have three candidate datasets available for a similarity ranking mechanism. We analyse each dataset separately due to their intrinsic differing semantics. Figure 2 gives an

[2]http://www.wikipedia.org/
[3]http://wiki.dbpedia.org/Datasets
[4]Version 3.8, crawled June 2012
[5]blank nodes are placeholders for entities but cannot actually be resolved in a graph
[6]http://en.wikipedia.org/wiki/Help:Infobox
[7]http://wiki.dbpedia.org/Downloads38

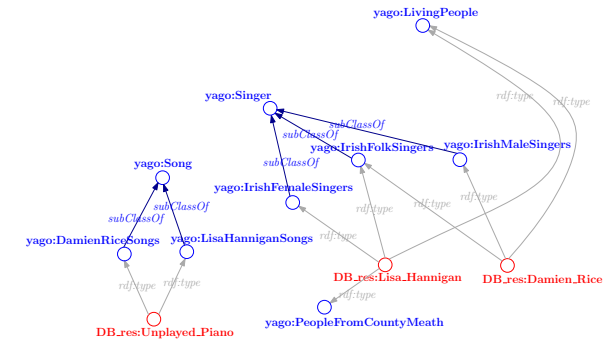
example of the DBpedia graph structure of Infobox, Categories, YAGO and is serves an illustrative example to give an intuition of the differences between the different graphs.



(a) Infobox graph example



(b) SKOS category graph example



(c) YAGO graph example

Figure 2: Prototypical examples of Infobox, Category, and YAGO graphs. Red nodes are our nodes of interest, Lisa Hannigan and her known related entities (collaborator Damien Rice and duet song Unplayed Piano), blue nodes and edges belong to the corresponding ontology.

For the problem of faceted search, what properties of these networks would be useful for ranking and clustering? In Figure 2 we have Lisa Hannigan as the central resource node, with Damien Rice and their duet song Unplayed Piano also highlighted. We can see that the Infobox graph appears more connected in that it is possible for the three (clearly related) resources to reach each other within an edge hop. However, the structure is very different compared to the other two (stricter) ontological graphs, of which amongst themselves are somewhat similar. In all three graphs the general structure is that all resource nodes are connected to the corresponding ontology via the *rdf:type* or *purl:subject* edge. The highlighted blue nodes and edges then show the hierarchical structure of how concepts in the ontologies are linked.

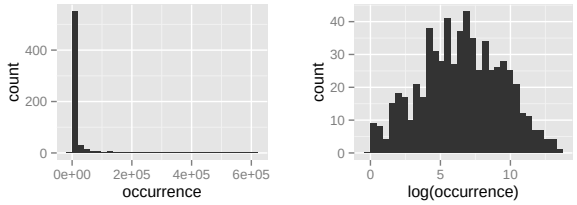


Figure 3: Infobox frequency of type.

	# SCC	largest cpt size	# WCC	largest cpt size
Infobox	2,015,629	120,833	6,844	2,262,130
Categories	4,679,423	10,698	1,105	4,694,520

Table 1: Strongly (SCC) and Weakly (WCC) connected components of combined graphs.

Table 2 shows the statistics of the datasets distinguished by their node types. The resource nodes (res) map directly to a Wikipedia article. They are the red nodes in Figure 2, and the black and grey edges correspond to the res edges. Similarly, the blue nodes and edges in Figure 2 make up the ontology/Category graphs and are the ont nodes and edges in Table 2.

Combining this with the cursory view of the datasets in Figure 2 we can see that our likely candidates are the richly connected Infobox properties for similarity ranking and the ontology (blue nodes and edges) networks for cluster labelling. We investigate the connectivity of the concepts by looking at the connected components of each of the graphs (as a whole, with the corresponding ontologies/category.). We can see from Table 1 that there are many strongly connected components in both the Infobox and Category datasets.[8] In particular we observe that the Categories dataset has a rather large strongly connected component which confirms that there exists at least one large cycle. We note that this will need to be taken into account in our discussion for future work.

6.1 Infobox Properties for Similarity Ranking

Thus, we use the Infobox properties graph for similarity ranking because of its strong connectivity and large variation of semantic edge types. We can see in Figure 3 that the distribution of the occurrence of the different types (properties) found in the DBpedia dataset is far from uniform. The top 5 edge types are *country, team, isPartOf, birthPlace* and *genre* which can be semantically interpreted as quite broad terms, whereas some of the less frequent edge types such

[8]Note that due to the fact that we later find that its coverage of the resources for the infobox dataset is too low to be of practical use, we have omitted the YAGO dataset from this analysis as it adds no value.

Dataset	# res nodes	# res edges	# edge types	# ont nodes	# ont edges	# edge types
Infobox Ontology	1,905,941	6,157,465	1	359	359	1
Categories	4,534,546	15,112,372	1	762,025	1,730,458	2
YAGO	8,167,440	17,594,297	1	4,404,682	4,404,682	1
Infobox properties	2,283,173	9,743,942	629	n/a	n/a	n/a

Table 2: Node and edge types of each of the datasets. # res are the number of nodes corresponding to Wikipedia articles, # ont correspond to nodes defined on the ontology/Category, # types are the number of distinct edge types.

as *firstFlight, lastFlight, associateEditor, vicePrimeMinister* and *riverBranchOf* are more specific terms and therefore we employ an approach commonly found in the text mining literature, TF-IDF, for weighting the edges, which we describe in detail in Section 7.1.

6.2 Ontologies for Labelling

We use the Infobox properties for ranking due to the rich connections available between the resources on Wikipedia. After we retrieve nodes that are ranked similar to the query we need to be able to map the results back to some dataset for labelling. We found that there were in fact 262,928 Wikipedia resources in the Infobox dataset that were not found in any of the three ontology/Categories datasets. Therefore, the maximum % coverage shown in Table 3 is the maximum *possible* coverage of all the remaining resources that can be covered by any of the other datasets. The results show that surprisingly, it is the Wikipedia Category has highest coverage (higher than the DBpedia Ontology), at 99.42%. Furthermore, by combining with the DBpedia Ontology, we gain a further 11,462 resources resulting in a 99.92% overall coverage and so we use these two datasets as our candidates for labelling clusters.[9]

	raw number	raw %	maximum %
Wikipedia Categories	2,007,064	87.907	99.423
DBpedia Ontology	1,811,451	79.339	90.855
YAGO	1,326,570	58.102	69.618

Table 3: Coverage of ontologies over Infobox properties resource nodes.

7. EXPLORATORY SEARCH METHODS

The following sections describes in detail the methods we use for ranking, clustering and labelling.

7.1 Ranking

We are given a query node q and we would like to find nodes that are similar to it. Here, our links contribute to similarity, so e.g., if a path connects two nodes we assume that they are similar. We assume this because in a semantic graph the predicates are defined by relationships between objects and subjects. So if v_1 is related to v_2 and v_2 is related to v_3, then v_1 is related to v_3 via v_2, i.e. we make an assumption that similarity, in terms of relatedness, is transitive in nature. In short, paths are a chain of relations, relations are important for similarity, thus paths give an indication of relatedness.

The k-step (in our case, l) Markov centrality [20] is known to find nodes that are relatively central to a given query node, which is what we use it to quantify relatedness. We restrict the step size to 2 because we found that the number of items retrieved for $l \geq 3$ become largely irrelevent to the query.

To take the link types into account, we apply a scaled weighting according to the frequency of the edge types (leaving the uniform case as a comparable baseline). We define our weighting scheme by drawing inspiration from the TF-IDF approach commonly used in text mining such that we give more weight to 'rare' types (analagous to terms) in the network (analogous to the documents). This is based on the intuition that rarer link types is more informative in similarity calculation, e.g. the fact that the link type *associatedArtist* occurs less frequently than *birthPlace* across the whole infobox dataset means that it is more important for similarity, and so we weight it higher. Formally, we have

$$t\_inv_i = \frac{1}{|t_i|} \tag{1}$$

where $|t_i|$ is the number of occurrence of type i from the infobox ontology in the whole graph, e.g. in Figure 2a the occurrence of type *musicalArtist* is 2, *associatedBand* is 3 etc. Thus the weight, or probability, from node j to i is:

$$prob(j,i) = w_{ji} = \frac{freq(t_i)}{\sum_{m=1}^{deg(j)} t\_inv_m} \tag{2}$$

$$\sum_{i=1}^{deg(j)} prob(j,i) = 1 \tag{3}$$

where $freq(t_i)$ is the frequency of type t_i in the neighbours of node j, $N(j)$, again analogous to the single document in the TF-IDF framework.

We effectively assign a weighted probability which combines the global frequency occurrence of the type scaled by its local frequency in the neighbours of the node.

The ranking for nodes of up to two hops away from a is then:

$$rank(a,b) = \tilde{p}(a,b) + \tilde{p}(a,k) * \tilde{p}(k,b), \forall k \in N(a) \tag{4}$$

where $\tilde{p}(a,b)$ is the Monte Carlo approximation of $prob(a,b)$.

Thus, for query node q:

- Run random walk up to length $l = 2$ $n = 1000$ times.
- For each run, count the nodes that were 'hit'.
- Rank the nodes according to the number of times they were hit, such that the higher the hit, the higher the rank.

We choose to fix $n = 1000$ as we found that this was an appropriate number of iterations for most nodes since we found that the degree distribution was a long-tailed one. We address the problem for nodes that have many neighbours, such as USA (which has $\approx$ 220K) in our discussion.

[9]We note that tried the other combinations as well and this was the optimal combination-of-two, omitting YAGO due to its overall poor and low additional coverage gain.

7.2 Clustering

After we have obtained a list of rankings of related entities to our query node q, we need to cluster them into meaningful groups as this is what we hope to achieve as part of the exploratory search. For this, we again take the TF-IDF mindset and use edge types linking to immediate neighbours of the ranked nodes as features, analagous to term vector in text clustering.

For each node in the ranked list obtained from the query, we create a set of expected terms as the set union of all the types of the neighbours of each node in the list, T. Thus for each node in the list its vector contains the frequency of its neighbour's type over the space of T. We use the cosine distance for similarity measurement between the ranked nodes' vectors and use Ward's minimum variance to merge clusters using hierarchical agglomorative clustering [19].

Cluster Labelling

As shown in Section 6, Table 3, we use the Wikipedia Category and DBpedia Ontology as candidates for labelling because their combination gives greatest coverage of the Infobox resource nodes. For each cluster, we show the top 3 overlapping nodes up to two hops in the Ontology/Category. We are currently investigating other methods for this labelling such as extending the relatedness measures in [12] for more than two entities.

8. RESULTS

For comparison, we show results for the query Lisa Hannigan obtained using the weighted and unweighted approach, which we refer to as INVERSE and UNIFORM, respectively. Table 4 shows the comparison of the clusters obtained from the inverse weighted scheme versus those found using the unweighted scheme. The clusters are arranged so that they are approximately equivalent in labels. Table 5 shows the top 3 labels found using the Wikipedia Categories and DBpedia Ontology.

Lisa Hannigan is an Irish singer-songerwriter who collaborated extensively with Damien Rice in the early stages of her career. The first cluster appears to be the most coherent one, consisting of instruments which is well described by the labels found in the Wikipedia Categories, and not available from the DBpedia Ontology, for both INVERSE and UNIFORM.

From manual inspection, the second cluster mainly contains songs of related artists for INVERSE whereas UNIFORM retrieves a bigger cluster with albums. We see that the labels found from the Categories data for this cluster is more specific and informative than the DBpedia Ontology, where its best label only broadly describes the cluster as MusicalWork. The Categories labels show the differences between the clusters better, with songs appearing in all three labels for INVERSE, and a mixture of artist and albums for UNIFORM.

Cluster 3 for INVERSE consist of albums only, and the UNIFORM cluster is more varied with songs, music genres and albums appearing. This is despite the size of the INVERSE cluster being much bigger than the UNIFORM one. Again the labels in Table 5 show that the Categories labels are more informative, particularly for the INVERSE case although the the DBpedia Ontology label, MusicalWork, can be seen as a good label for the heterogeneous UNIFORM cluster.

The fourth and fifth clusters for both methods are similarly varied, with no one clear description that would link all the items. This is especially reflected in the unrelated top three labels found from the Categories labelling for cluster 5. In this case it is perhaps the broad description from the DBpedia Ontology that is more useful, using MusicalWork to describe cluster 4 of INVERSE and Organisation for UNIFORM. It is also noted that while cluster 5 consists of varying entities, for both cases there is in fact a significant enough number of items that are location that explains the DBpedia Ontology labels of Place and PopulatedPlace.

The final cluster for both methods clearly consists of related musicians and bands. However, this is not sufficiently described by the labels found using the Categories labelling, and the label of Artist from the DBpedia Ontology is perhaps the most informative one. This is likely due to the fact that in our labelling process we are retrieving the labels for up to two hops away from the original items, and since every artist is an instance of a person, this is obviously a 'good' candidate label by this heuristic.

Another interesting result we found was that the number features used for clustering for INVERSE was generally lower than UNIFORM, e.g., we found that the average number of terms found by selecting 100 random nodes was 376.31 (s.d. 306.97) for INVERSE versus 455.78 (s.d. 297.1) for UNIFORM for the same nodes. Combining this finding with the overall more cohesive clusters we found using INVERSE, this indicates that the INVERSE approach is a better candidate for finding results that can be clustered according to a better feature space.

9. CONCLUSION

We now can answer the questions we posed as part of our motivation:

Question Do subsets of DBpedia differ from each other?

Answer Yes, we have shown that the ontologies associated with each dataset (subset) of DBpedia can be represented as a heterogeneous graph and as such, we found that the DBpedia ontology, derived from the infoboxes, to have a unique structure from the other two (Categories & YAGO) in terms of connecting the Wikipedia articles.

Question Can we provide more effective similarity ranking process in light of these findings?

Answer We use the finding of non-uniform distribution of the Infobox properties to weight our edges to compute a Markov centrality that ranks similar nodes to a given query node. Then we make use of the descriptive and higher coverage datasets – Categories and the DBpedia ontology itself – to provide labels of clusters that are found from using the Infobox properties as features.

Overall we have presented a framework for conducting an exploratory search on Wikipedia using the underlying graph structure of DBpedia. We do this by assessing the different datasets of DBpedia as graphs and in particular, exploit

	INVERSE		UNIFORM	
Cluster #	**Cluster elements**	$\|k\|$	**Cluster elements**	$\|k\|$
1	Singing, Cello, Violin, Clarinet, Rhodes_piano,Guitar, Keyboard_instrument, Piano, Synthesizer, Wurlitzer	17	Singing, Cello, Wurlitzer, Keytar, Fender_Telecaster_Deluxe, Gretsch_6120, Piano, Synthesizer, Ukulele, Mandolin	17
2	...In_Translation, Called_Out_in_the_Dark, The_Remedy_(I_Won't_Worry), I'm_Yours_(Jason_Mraz_song), I_Won't_Give_Up, Make_It_Mine, Dogs_(Damien_Rice_song), Wordplay_(song), You_and_I_Both, Rootless_Tree	21	I_Will_Follow_You_into_the_Dark, Called_Out_in_the_Dark, Wordplay_(song), Make_It_Mine, Sing_the_Changes, It's_Amazing, Passenger_(Lisa_Hannigan_album), Death_By_Stereo_(album), 9_Crimes, Just_Say_Yes_(song)	87
3	Late_Night_Tales:_Snow_Patrol, O_(Damien_Rice_album), Flood_(Herbie_Hancock_album), Sea_Sew, Fat_Albert_Rotunda, Yours_Truly:_The_I'm_Yours_Collection, Jazz_Africa, CoreaHancock, Passenger_(Lisa_Hannigan_album), Not_Fade_Away_(David_Kitt_album)	54	Down_to_Earth_(Jem_album), The_Trip:_Created_by_Snow_Patrol, Okonokos, Independent_music, Bebop, Folk_music, Indie_pop, Power_pop, Alternative_country, Lille_(song)	21
4	Indie_folk, Rock_music, Folk_music, Woman_Like_a_Man, Lille_(song), Birdtalk, Skylarkin'_(Mic_Christopher_album), Bloodless_Coup, The_Roads_Outgrown, Universal_Island_Records	25	Called_Out_in_the_Dark, Wordplay_(song), Make_It_Mine, Sing_the_Changes, It's_Amazing, Count_Me_In_(311_song), The_Remedy_(I_Won't_Worry), 9_Crimes, Just_Say_Yes_(song)	61
5	County_Meath, Blarney, Kilcloon, Leinster, United_States, Netherlands, Republic_of_Ireland, Mic_Christopher, Alice_Stopford_Green, Singer-songwriter	40	County_Meath, Blarney, Western_European_Time, Leinster, Western_European_Summer_Time, Kilcloon, Music_download, Compact_Disc_single, John_Carney_(director), Round_Midnight_(film)	54
6	Jason_Mraz, Jack_Johnson_(musician), Bell_X1_(band), Snow_Patrol, Damien_Rice, Belle_&_Sebastian, Joni_Mitchell, Josh_Ritter, Songs_By_Damien_Rice, The_Cake_Sale	59	Vyvienne_Long, Songs_By_Damien_Rice, Tom_Smith_(musician), Nina_Persson, Peter_Buck, Gary_Lightbody, Cathy_Davey, Juniper_(band), Fleet_Foxes, The_Swell_Season	45

Table 4: Sample query and cluster results with labels: inverse and uniform weights, $|k|$ is the size of the cluster. For brevity, we show the first ten results.

	INVERSE		UNIFORM	
Cluster #	**Wikipedia Categories**	**DBpedia Ontology**	**Wikipedia Categories**	**DBpedia Ontology**
1	Musical_instruments, String_instruments, Percussion	N/A	Musical_instruments, String_instruments, Percussion	N/A
2	Songs_by_artist, Jason_Mraz_songs, American_pop_songs	Work, MusicalWork, Single	Albums_by_artist, Songs_by_artist, Herbie_Hancock_albums	Work, MusicalWork, Thing
3	Albums_by_artist, Herbie_Hancock_albums, Jazz_albums_by_American_ artists	MusicalWork, Work, Thing	Musical_subgenres_by_genre, Rock_music_genres, Music_genres	Thing, MusicGenre, MusicalWork
4	Songs_by_artist, Missing_people, Date_of_birth_unknown	MusicalWork, Work, Thing	Missing_people, Year_of_death_missing, Dead_people	Agent, Thing, Organisation
5	Albums_by_artist, Member_states_by_organization, Jazz_albums_by_American_ artists	Place, PopulatedPlace,Thing	Towns_and_villages_in_the_Republic_of_Ireland_by_county, Towns_and_villages_in_Ireland_by_county, Geography_of_County_Meath	PopulatedPlace, Place, Settlement
6	Place_of_birth_missing_(living_people), Missing_people, Year_of_death_missing	Agent, Artist, Person	Place_of_birth_missing_(living_people), Missing_people, Year_of_death_missing	Agent, Artist, Person

Table 5: For each cluster, the top three labels from the Wikipedia Category and DBpedia Ontology (where available).

the richly heterogeneous nature of the types in the DBpedia properties graph to employ a ranking and clustering procedure inspired by the text mining domain's TF-IDF method. Lastly, we showed some concrete results by using the sample query of the singer-songwriter Lisa Hannigan to show that the clusters found using the weighted method of clustering performs better in finding more coherently labelled groups than the unweighted method.

One of the main advantages of our ranking method is that it is very scalable. The Monte Carlo estimation can be done in parallel as the estimations are independent of each other. The feature space for clustering is generally small (within the hundreds) and so the computation for hierarchical agglomorative clustering is not intensive. Overall our framework takes advantage of both the network structure and intuition that rare types/features are more important for similarity calculation in a straightforward and scalable manner.

We are also aware that this approach has its drawbacks, for instance, it is not applicable to networks where the edge type distribution is a uniform. Also, in terms of our Wikipedia exploratory search scenario, the process is only currently applicable to nodes that have an infobox.

In summary, we have presented a first approach to computing similarity in an exploratory search context, that is applied on an a reasonably large dataset with some promising initial results. Lastly, we present some discussion for future work to build upon these results.

10. DISCUSSION

Some technical aspects in this work that can be improved involve the choice of iteration parameter and refinement of clusters. In the presented work we have currently fixed the iteration parameter to 1000 which works reasonably well for many nodes (as the degree distribution is a long-tailed one,

i.e. there are few nodes with *very* high degree, such as ones that are countries), but we would need to take into account these larger degree nodes. One approach may be to penalise certain link types once the degree is greater than some threshold. With regards to the cluster number, we would need to investigate further what would be an automatic method of selecting number of clusters which is an age-old question in itself. Particular to our problem is the need to balance between an optimisation method of clusters being 'close' in a dimensional space versus the capacity of people to comprehend an optimal number of concepts. Moreover, part of our current work is the planning of a case study where participants are given information retrieval tasks using the presented system versus some of the related works for comparison.

Another area of interest is to be able to extract sub-networks with fixed schema to compare with PathSim and other methods mentioned in the Related Work. For this, we would need to have a methodical way of extracting subsets with specific types for comparision. For this, we note that for each class in the DBpedia ontology, there are associated properties with each class which may be used[10], e.g. the properties associated with the Game class are restricted to only 5 types. However, the consistency of the properties is not always so clear, e.g. for the class of Food has a property *chancellor*, but when searched for its occurrence in the DBpedia Infobox dataset we only found its use in linking a person to a University (i.e. person X is chancellor of University Y).

Lastly, the presented work may well be applicable to other knowledge networks where distribution of link types is not uniform, e.g. an interesting future work could be rather than using just the pure page links of webpages, one can include the text of the link as the link type information.

Acknowledgements

Research presented in this paper was supported by Science Foundation Ireland under Grant No. 08/SRC/I1407 (Clique: Graph & Network Analysis Cluster) and Grant No. SFI/08/CE/I1380 (Lion-2). We would like to thank Jeffrey Chan, Václav Belák and Donn Morrison for providing many helpful suggestions and comments.

11. REFERENCES

[1] C. Bizer, J. Lehmann, G. Kobilarov, S. Auer, C. Becker, R. Cyganiak, and S. Hellmann. Dbpedia-a crystallization point for the web of data. *Web Semantics: Science, Services and Agents on the World Wide Web*, 7(3):154–165, 2009.

[2] R. Hahn, C. Bizer, C. Sahnwaldt, C. Herta, S. Robinson, M. Bürgle, H. Düwiger, and U. Scheel. Faceted wikipedia search. In *Business Information Systems*, pages 1–11. Springer, 2010.

[3] M. A. Hearst. Clustering versus faceted categories for information exploration. *Communications of the ACM*, 49(4):59–61, 2006.

[4] P. Heim, S. Hellmann, J. Lehmann, S. Lohmann, and T. Stegemann. Relfinder: Revealing relationships in rdf knowledge bases. *Semantic Multimedia*, pages 182–187, 2009.

[5] I. Hulpus, C. Hayes, M. Karnstedt, and D. Greene. Unsupervised graph-based topic labelling using dbpedia. In *Proceedings of the sixth ACM international conference on Web search and data mining*, pages 465–474. ACM, 2013.

[6] J. Leal, V. Rodrigues, and R. Queirós. Computing semantic relatedness using dbpedia. *In Proceedings SLATE 2012*, page 3519, 2012.

[7] S. Lee, S. Park, M. Kahng, and S.-g. Lee. Pathrank: a novel node ranking measure on a heterogeneous graph for recommender systems. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 1637–1641. ACM, 2012.

[8] C. Li, N. Yan, S. B. Roy, L. Lisham, and G. Das. Facetedpedia: dynamic generation of query-dependent faceted interfaces for wikipedia. In *Proceedings of the 19th international conference on World wide web*, pages 651–660. ACM, 2010.

[9] E. Minkov and W. W. Cohen. Learning to rank typed graph walks: Local and global approaches. In *Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis*, pages 1–8. ACM, 2007.

[10] R. Mirizzi, T. Di Noia, A. Ragone, V. Ostuni, and E. Di Sciascio. Movie recommendation with dbpedia. In *3rd Italian Information Retrieval Workshop (IIR 2012). CEUR-WS*, 2012.

[11] R. Mirizzi, A. Ragone, T. Di Noia, and E. Di Sciascio. Ranking the linked data: the case of dbpedia. In *Web Engineering*, pages 337–354. Springer, 2010.

[12] A. Panchenko. A study of heterogeneous similarity measures for semantic relation extraction. In *Proceedings of the JEP-TALN-RECITAL*, volume 3, pages 29–42. ATALA & AFCP, 2012.

[13] A. Passant. Measuring semantic distance on linking data and using it for resources recommendations. In *Linked AI: AAAI Spring Symposium Linked Data Meets Artificial Intelligence. AIII*, 2010.

[14] C. Ramakrishnan, W. H. Milnor, M. Perry, and A. P. Sheth. Discovering informative connection subgraphs in multi-relational graphs. *ACM SIGKDD Explorations Newsletter*, 7(2):56–63, 2005.

[15] P. Schonhofen. Identifying document topics using the wikipedia category network. In *Web Intelligence, 2006. WI 2006. IEEE/WIC/ACM International Conference on*, pages 456–462. IEEE, 2006.

[16] Y. Sun, J. Han, X. Yan, P. Yu, and T. Wu. Pathsim: Meta path-based top-k similarity search in heterogeneous information networks. *VLDB*, 2011.

[17] Y. Sun, J. Han, P. Zhao, Z. Yin, H. Cheng, and T. Wu. Rankclus: integrating clustering with ranking for heterogeneous information network analysis. In *Proceedings of the 12th International Conference on Extending Database Technology: Advances in Database Technology*, pages 565–576. ACM, 2009.

[18] Y. Sun, Y. Yu, and J. Han. Ranking-based clustering of heterogeneous information networks with star network schema. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 797–806. ACM, 2009.

[19] J. H. Ward Jr. Hierarchical grouping to optimize an objective function. *Journal of the American statistical association*, 58(301):236–244, 1963.

[20] S. White and P. Smyth. Algorithms for estimating relative importance in networks. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 266–275. ACM, 2003.

[21] N. Yan, C. Li, S. B. Roy, R. Ramegowda, and G. Das. Facetedpedia: enabling query-dependent faceted search for wikipedia. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 1927–1928. ACM, 2010.

[22] T. Zesch and I. Gurevych. Analysis of the wikipedia category graph for nlp applications. In *Proceedings of the TextGraphs-2 Workshop (NAACL-HLT 2007)*, pages 1–8, 2007.

[10]The list of classes and their properties can be found at: http://mappings.dbpedia.org/server/ontology/classes/