

A Hybrid Approach to Unsupervised Relation Discovery Based on Linguistic Analysis and Semantic Typing

Zareen Syed

University of Maryland Baltimore County
1000 Hilltop Circle
Baltimore, MD 21229, USA

Evelyne Viegas

Microsoft Research
One Microsoft Way
Redmond, WA 98052, USA

Abstract

This paper describes a hybrid approach for unsupervised and unrestricted relation discovery between entities using output from linguistic analysis and semantic typing information from a knowledge base. We use Factz (encoded as subject, predicate and object triples) produced by Powerset as a result of linguistic analysis. A particular relation may be expressed in a variety of ways in text and hence have multiple facts associated with it. We present an unsupervised approach for collapsing multiple facts which represent the same kind of semantic relation between entities. Then a label is selected for the relation based on the input facts and entropy based label ranking of context words. Finally, we demonstrate relation discovery between entities at different levels of abstraction by leveraging semantic typing information from a knowledge base.

1 Introduction

There are a number of challenges involved when using facts extracted from text to enrich a knowledge base (KB) with semantic relations between entities: co-reference resolution as there are many co-referent objects; entity resolution in order to link the entities mentioned in text to the right entities in the KB; handling co-referent relations, as a particular semantic relation between entities can be expressed in a variety of ways in the text and therefore have multiple facts associated between the entities. In addition, the facts extracted from linguistic analysis are usually noisy and sparse.

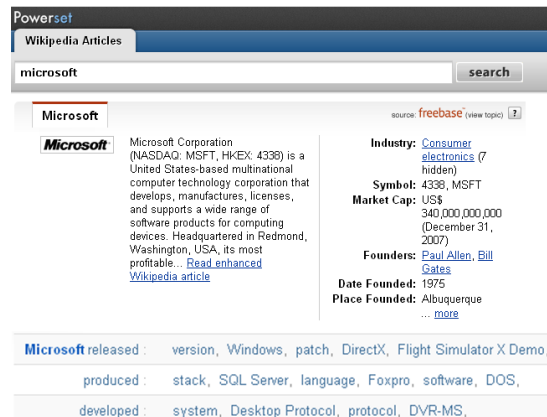
Our work focuses on a recent line of exploratory work in the direction of Unrestricted Relation Discovery which is defined as: the automatic identification of different relations in text without specifying a relation or set of relations in advance (Shinyama and Sekine, 2006). We use the facts which are the output of linguistic analysis from Powerset (www.Powerset.com). Powerset

is an online search engine for querying Wikipedia using Natural Language Queries. Powerset performs a linguistic analysis of the sentences within Wikipedia and outputs facts in the form of subject, predicate and object triples which can be queried through the online interface. For most entities like persons, places and things, Powerset shows a summary of facts from across Wikipedia (figure 1). In our approach we use the readily available “Factz” from Powerset as input to our system. Powerset is Wikipedia independent and can run on any corpus with well-formed sentences and hence our approach is also not limited to Wikipedia. The Factz output from Powerset may represent relations between named entities or just nouns for example,

Bank of America <acquired> bank
Bank of America <acquired> Merrill Lynch
Bank of America <owned> building

Linguistic analysis has been recently described as an effective technique for relation extraction (Yan et al., 2009; Kambhatla, 2004; Nguyen et al., 2007). Following that trend, we incorporate Factz, that are the output of linguistic analysis done by Powerset, to discover semantic relations between entities.

Information from existing knowledge re-



The screenshot shows the Powerset interface for the query 'microsoft'. It displays a summary of facts for Microsoft Corporation, including its industry (Consumer electronics), symbol (4338, MSFT), market cap (US\$ 340,000,000,000), and founders (Paul Allen, Bill Gates). Below the summary, there are several Factz entries, such as 'Microsoft released: version, Windows, patch, DirectX, Flight Simulator X Demo', 'produced: stack, SQL Server, language, Foxpro, software, DOS', and 'developed: system, Desktop Protocol, protocol, DVR-MS'.

Figure 1. Demonstration of Powerset Factz available online

sources can help in tasks like named entity disambiguation by providing additional context in the form of linked entities in the KB and aid in linking the entities mentioned in the text to the entities in the KB. The KB can also provide information about the entity types which can in turn be used to discover relations between entity types at different levels of abstraction and help in enriching the KB itself. This could allow ontology engineers to explore the kind of relations existing between different entity types in a corpus and then design an ontology which is representative of the entities and relations evident in the corpus.

Our overall approach to automatic relation discovery consists in a hybrid approach based on Powerset Factz that are the output of linguistic analysis, and serve as input to our system; Text based label ranking by directly considering the context words in the sentences; and, Semantic Typing information from existing knowledge resources to discover relations between Entity types at different levels of abstraction.

The paper is organized as follows. We discuss the related work in the next section. In section 3 we propose our approach and give the details of different components in our system. In section 4, we discuss preliminary experiments and results. In the last section we conclude our work and give future work directions.

2 Related Work

Hasegawa et al. (2004) developed an approach for unsupervised relation discovery by clustering pairs of entities based on intervening words represented as context vectors. They used the most frequent common word to label the cluster and hence the relation represented by the cluster.

Shinyama and Sekine (2006) developed an approach to preemptively discover relations in a corpus and present them as tables with all the entity pairs in the table having the same relations between them. For pairs of entities they generate basic patterns that are parts of text syntactically connected to the Entity and use the predicate argument structure to make the basic patterns more generalized. They generate a basic cluster from articles based on having similar basic patterns to represent the same event and then they cluster the basic clusters to get a set of events having the same relation.

Davidov et al. (2007) developed a web mining approach for discovering relations in which a specified concept participates based on clustering patterns in which the concept words and other words appear. Their system is based on the initial seed of two or more words representing the type of concept one is interested in.

Linguistic analysis has been reported as an effective technique for semantic relation extraction. Harabagiu et al. (2005) used shallow semantic parsers to enhance dependency tree kernels and to build semantic dependency structures to improve relation extraction, they reported that their method improved the quality of the extracted relations as compared to kernel-based models that used semantic class information only.

Nguyen et al. (2007) presented an approach for relation extraction from Wikipedia by extracting features from subtrees mined from the syntactic structure of text. Kambhatla (2004) developed a method for extracting relations by applying Maximum Entropy models to combine lexical, syntactic and semantic features and report that they obtain improvement in results when they combine variety of features. Most of the existing approaches have used linguistic analysis to generate features for supervised or semi-supervised relation extraction.

Recently, Yan et al. (2009) have developed an approach for unsupervised relation discovery by integrating linguistic analysis done on Wikipedia with context generated from the Web. They develop a clustering approach based on dependency patterns from dependency analysis of Wikipedia and surface patterns by querying the web to introduce redundancy. They report that dependency patterns improve the precision whereas, the surface patterns improved the coverage.

Banko et al. (2008) introduce the TextRunner system which takes a small corpus sample as input and uses a linguistic parser to generate training data which they use to train the extractor which can run at web scale. However, Kok and Domingos (2008) have reported that the triples output from the TextRunner system are noisy, sparse and contain many co-referent objects and relations which is also the case with Powerset. Their system uses the output from the TextRunner system and uses Multiple Relational Clustering model to get object clusters and relation clusters.

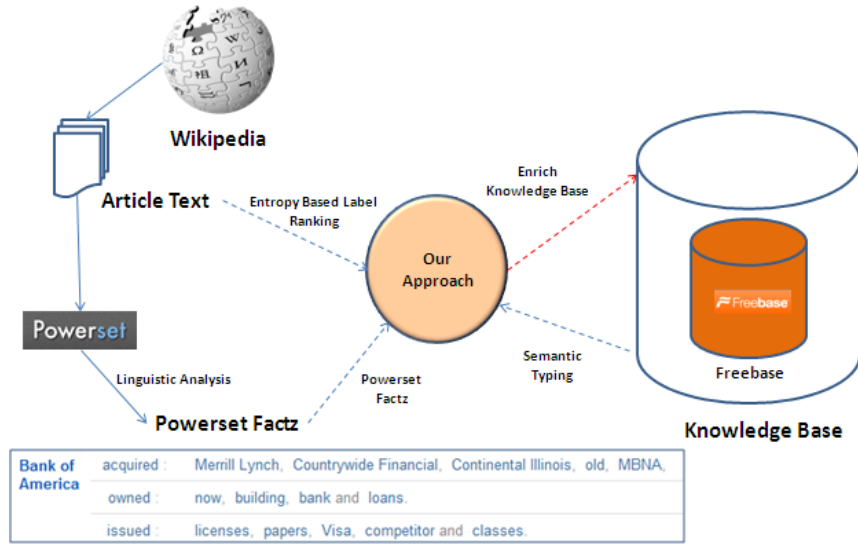


Figure 2. The Knowledge Discovery approach uses Powerset Factz which are the output from linguistic analysis, article text for entropy based label ranking and existing knowledge resources for discovering relations at different levels of abstraction and hence aiding in enriching the existing knowledge resources.

3 Approach

In this section we describe in detail the different steps in our approach involving querying Factz from Powerset, collapsing facts expressing same type of relation, Label Selection and introducing Semantic Typing information. Figure 2 gives an overview of our approach and Figure 3 shows the different components in our system. We discuss each component in detail below.

3.1 Querying Powerset and Retrieving Factz

In the first step we query Powerset API by giving as input a list of entities or list of entity pairs and retrieve all the Factz and sentences that are associated with the entities or entity pairs from the Powerset API output.

3.2 Collapsing Similar Relations

A particular semantic relationship can be expressed in different ways in sentences. For example words like “purchase”, “buy” and “acquire” may represent the same semantic relation between the subject and the object. Sometimes the words might be direct synonyms in which case resources like WordNet (Miller et al., 1990) can help in identifying the same relation whereas in other cases the words might not be synonyms at all but may still imply the same semantic relation between the subject and the object. For example, we queried Powerset to get a sample of relations between companies and products. We

got relations like *introduce*, *produce*, *sell*, *manufacture* and *make*. It is often the case that companies *introduce* and *sell* the products that they *manufacture*, *make* or *produce*. However, all of these words are not synonyms of each other and it may not be feasible to express the relation between a company and a product in all these different ways in a KB.

We have developed an approach for collapsing relations expressed using different words in the facts and represent it using the dominating relation between the pair of entities. We explain the different steps in our approach below.

3.2.1 Relation Clustering

We consider relations to be similar if they appear between the same subjects and the objects. We take the set of Factz that we got by querying Po-

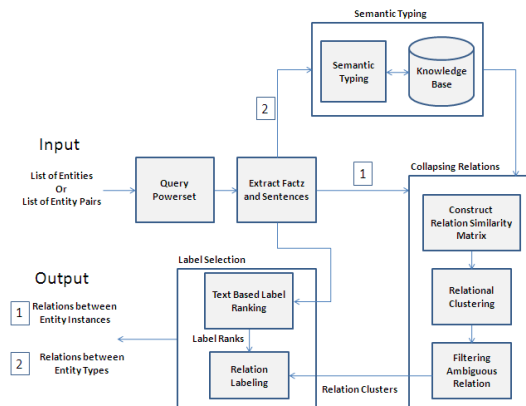


Figure 3. System Framework

werset in the previous step and based on those Factz we construct a similarity matrix to represent similarity between all pairs of relations in the data set. Each entry in the similarity matrix represents the number of times the pair of relations had the same subject and object in the Factz data set. For example, in the sample dataset in table 1, the similarity matrix entry for the pair *acquired* and *purchased* would be 3. We use that similarity matrix as input and apply average link agglomerative clustering algorithm over it.

Subject	Predicate	Object
Bank of America	acquired	Merrill Lynch
Bank of America	acquired	MBNA
Bank of America	acquired	FleetBoston
Bank of America	purchased	FleetBoston
Bank of America	purchased	Merrill Lynch
Bank of America	purchased	MBNA

Table 1. Relations between same subjects and objects in Powerset

3.2.2 Filtering Ambiguous Relations

After the clustering step we have a step for filtering ambiguous relations from the clusters. We explain the filtering procedure using an example from one of the experiments in which two clusters were produced. First cluster had *acquire*, *purchase*, *buy* and *own* relations and the second cluster had *introduce*, *produce*, *make* and *say about* relations. After clustering the relations we have the following steps:

1. We take each pair of entities and get the set of relations between the pair of entities. For example, the set of relation between “Bank of America” and “Merrill Lynch” are *acquire*, *purchase* and *say about* (figure 4).

2. By considering the set of relations between each pair of entities we assign it to a cluster based on the maximum number of overlapping relations between the set and the cluster members. In our example clusters, we assign it to cluster one with which there is an overlap of two relations i.e. *acquire* and *buy* instead of assigning it to cluster two with which it has an overlap of one relation i.e. *say about* (figure 4).

3. Once an entity pair is assigned to a cluster, we consider other relations in the set of relations present between that entity pair and if any of those relations exists as a member of another cluster we filter out that relation from that cluster. For example, one of the relations present between “Bank of America” and “Merill Lynch” is

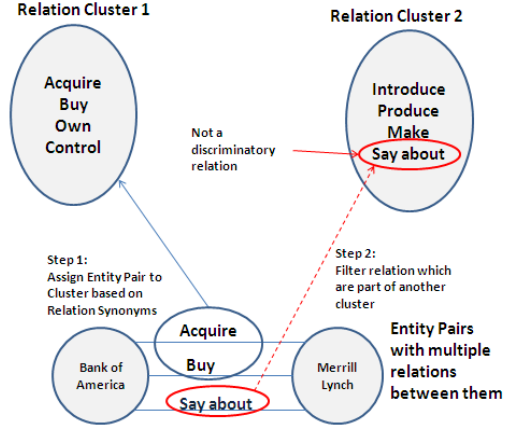


Figure 4. Filtering ambiguous relations from existing clusters

say about, and this relation is actually a member of cluster two whereas, this pair is assigned to cluster one and therefore, we filter out *say about* from cluster two. After cluster filtering, the label for the cluster is selected as the label that is the most frequent relation found in the set of entity pairs being assigned to the cluster.

3.3 Relation Label Selection

A pair of entities might have more than one fact associated with them. We select a representative label based on a hybrid approach by combining the output from entropy based label ranking (Chen et al., 2005) and clusters of similar relations found by relational clustering. We select the relation label as the cluster label of the cluster which has the maximum member overlap with the predicates in the set of facts between a pair of entities. In case there is an overlap of just one relation, we select the label that is ranked highest through entropy based label ranking approach (Chen et al., 2005). According to their algorithm, the importance of terms can be assessed using the entropy criterion, which is based on the assumption that a term is irrelevant if its presence obscures the separability of the dataset. There may be cases where there are multiple relations existing between a given pair of entities, however, in our approach we select the relation label that is evident in the majority of the facts associated with the pair.

3.4 Semantic Typing

For certain applications there might be the need of discovering relations between specific types of entities rather than instances of entities. For example, for ontology engineering, the ontology

engineer might want to explore the kind of relations that exist between different entity types based on the data set and then develop an ontology representing those relations. Therefore, we have a component in our system that incorporates semantic type information into the Factz before collapsing the relations present in the facts. The semantic type module queries a knowledge base for the entity type and replaces the entity instance names with entity types in the Factz data set. We have used the Freebase (Metaweb Technologies, 2009) Knowledge base to associate the entity types for the entities that we experimented with. When this modified version of the Factz dataset is given as input to the next component of the system i.e. Collapse Relations, the similarity between relations is computed based on having the same subject and object entity types rather than entity instances. Following the Semantic Typing path in the system would output the relations discovered between types of entities. Introducing Semantic Typing information can also help in creating redundancy in the dataset and overcome the data sparseness problem. For example in case of relations such as *acquire* and *purchase* if we cannot get evidence of overlap in the subject and object in the Factz dataset then we cannot assign them any similarity score in the similarity matrix however, if we replace the instance names with instance types and consider the overlap between the instance types we can get more evidence about their similarity.

4 Experiments and Results

In this section, we present the preliminary experiments we conducted to evaluate the approach. We start by an initial evaluation of Powerset Factz by comparing them with ground truth and text based label ranking (Chen et al., 2005). We then use our approach to discover relations between different entity types. The details of the experiments and results are discussed below.

4.1 Preliminary Evaluation of Powerset Factz

Our first experiment was targeted towards a preliminary evaluation of the accuracy of Powerset Factz themselves and their performance when compared with ground truth and with Entropy based label ranking approach which does not use any linguistic analysis. To achieve this we took the “acquisitions” table from Freebase. The “acquisitions” table has a list of companies and their

Approach	Accuracy
Powerset Factz based approach	85%
Entropy based Label ranking	72%

Table 2. Comparison of Powerset Factz based approach and Entropy based label ranking

acquisitions. We considered the acquisitions table as ground truth as this information is either entered manually by contributors or imported from Wikipedia via DBpedia. We queried Powerset by giving the entity pairs as input and were able to retrieve Factz for 170 pairs out of 1107 entity pairs present in Freebase table. The number of pairs for which Powerset returned Factz is low because Powerset currently extracts Factz from well formed sentences and not semi-structured or structured information such as tables or info-boxes in Wikipedia and the *acquisition* relation is mostly expressed in the form of tables or lists in Wikipedia articles. We applied relational clustering and stopped clustering when the similarity between the clusters was less than 4. We identified one cluster (*acquire, purchase, buy*) having more than one member and got 146 relations labeled accurately i.e. 85% accuracy through our approach. We repeated the experiment using Entropy based label ranking approach (Chen et al., 2005). We were mainly focusing on relations that were expressed by verbs. We took all sentences between a pair of entities from which Powerset had extracted Factz. We extracted verbs from those sentences and ranked those verbs based on the entropy based label ranking approach and considered any of the labels matching with the cluster members (*acquire, purchase, buy*) as correct prediction. We compared the results with the ground truth and got the accuracy of 72% (table 2). Our preliminary experiment on the sample dataset demonstrated that the relation labels assigned by Powerset have reasonably high accuracy when compared with ground truth i.e. 85% and also give higher accuracy as compared to the entropy based label ranking approach for the sample data set.

4.2 Discovering Relations between Different Types of Entity Pairs

In this experiment we wanted to explore if our approach was successful in discovering relations existing between different types of entity pairs and clusters the pairs into separate clusters.

We constructed two datasets using Wikipedia page links between articles on entities namely Persons and Organizations. Using “person” type and “organization” type specified in Freebase,

we were able to construct a list of Wikipedia articles that were on Persons and Organizations. The Wikipedia article links served the purpose of finding out which organizations are related to which other organizations and which persons are related to which organizations. The first dataset represented relations between Organizations whereas the second dataset represented relations between Persons and Organizations. We applied relational clustering for collapsing similar relations and evaluated the output clusters at different thresholds to see if they represented relations between different types of entities. At stopping with a threshold of 2 we found the following two clusters having more than one member: one of the clusters represented the relations present between a pair of Organizations (*acquire, purchase, buy, own, say about, take over*) and the other cluster represented the relations between Persons and Organizations (*formed, found, lead*) (table 3). The experiment confirmed the effectiveness of clustering approach as it clusters relations between different kinds of entity pairs into different clusters.

Relations	Clusters
Org-Org	Cluster 1: <i>acquire, purchase, buy, own, say about, take over over</i>
Pers-Org	Cluster 2: <i>found, lead, form</i>

Table 3. Relations between different types of entity pairs are clustered into different clusters

4.3 Improving Recall

In this experiment we were interested in finding if Factz from Powerset can help in discovering relations between entities that are not present in resources like DBpedia and Freebase. We took a list of organization (with > 28,000 organization names from Freebase and an internal Knowledge Base) and retrieved Powerset Factz having those organizations as subjects. We performed relation clustering and output clusters at different thresholds. We selected the minimum threshold for which there were at least two clusters with more than one member. From the two clusters, one cluster had *manufacture, produce* and *make* relations and the second had *acquire, purchase, own, operate* and *buy* relations (table 4). Our intuition was that the first cluster represented relations between organizations and products. Therefore, we took the “company-products” table from Freebase and compared it with our dataset. However, we could only find an overlap of 3 subject object pairs. The second cluster had relations that we earlier found to exist between organizations

having the *acquisition* relation between them, therefore, we took the “acquisitions” table from Freebase and compared it against our dataset. Comparing the pairs with our list of organizations, we found 104 pairs that had an organization as a subject and an object. Out of those 104 pairs 97 pairs were assigned to cluster 2 and 7 pairs were assigned to cluster 1. When we compared those 97 pairs with Freebase “acquisition” table (which had 73 pairs of organizations that overlapped with our dataset) we found that 66 existed in the set and were therefore predicted correctly. We then inspected the rest of the pairs manually and found that there were 16 additional pairs that were predicted to have the *acquire* relation and which were not present in the Freebase table. Therefore, this approach helped in identifying 16 additional organization pairs having *acquisition* relation between them correctly.

Cluster	Cluster Members
1	<i>manufacture, produce, make</i>
2	<i>acquire, purchase, own, operate, buy</i>

Table 4. Clustering results for Relations having Organizations as subjects

Statistics	
No. of pairs in Freebase table	73
No. of discovered pairs matching Freebase	66
No. of additional pairs discovered	16
Total no. of correctly discovered pairs	82/104
Accurate Predictions %age	78%

Table 5. Evaluation results for improving recall by discovering additional entity pairs having the *acquisition* relation

Another observation worth mentioning is that the *acquisition* relation is represented mostly in the form of tables in Wikipedia whereas Powerset only processes information that is present in sentences. In spite of that, our approach was able to find new entity pairs from text that did not already exist in information extracted by other sources (table 5).

4.4 Discovering Relations at Different Levels of Abstraction

In this experiment we introduced Semantic Type information in the Factz data set to discover relations at different levels of abstraction i.e. between Entity Types at different levels (For example School or Organization, where School is a type of Organization).

We took a list of 13000 organizations for which we had their Organization Types available

from an internal KB and queried Powerset for Factz between all pairs of organizations and were able to retrieve more than 88,000 Factz. We passed on the Factz to the Semantic Typing module to replace the Organization names with their types. The Factz dataset with Semantic Type information was given as input for collapsing relations, where the similarity matrix was constructed based on the same subject and object types (rather than same subject and object instances), after which the clustering was performed. We evaluated the clusters at different stopping thresholds but the system did not generate any meaningful clusters. We then looked into the dataset and realized that a lot of noise was introduced into the system due to various organization names which were very ambiguous and replacing the ambiguous organization names with organization types had magnified the noise. For example, in our organizations list there is an organization with the name “Systems” which is of type “Medical Instrument Supplies”. It had the following fact related to it: <3d systems> <manufacture> <systems>. Replacing the organization name with the type resulted in the following fact i.e., <multimedia graphics software> <manufacture> <medical instruments supplies>. Such ambiguous names when replaced with wrong types further magnified the noise.

4.4.1 Resolving Ambiguity

As discussed, ambiguous organization names introduced noise and replacing them with organization types magnified the noise. Therefore, it was important to resolve the ambiguity in the names of entities before applying Semantic Typing. There are different approaches than can be used to recognize and disambiguate Named Entities, which we discuss below.

4.4.1.1 Named Entity Recognition

Powerset has Factz that are extracted from sentences. The Factz may be present between Named Entities or even just words in sentences. For example “Accord” is a name of a trade union and is also a word. Running Named Entity Recognition systems over the sentences from which the Factz have been extracted can help in identifying named entities and in eliminating such factz which are not between named entities. In general, the relation extraction systems have an initial step where they identify entities in sentences through NER systems and then discover relations between those entities.

Most of Named Entity Recognition and Disambiguation systems use the contextual information to disambiguate between entities. The contextual information could be words in the sentences or other entities in the sentences where the entity is mentioned. Having some evidence that two entities are related in some way can also help in eliminating much of the ambiguity. In general, the relation extraction systems have an initial step where they find related entity pairs based on Co-occurrences and then discover relations between those pairs of entities which frequently co-occur with each other in sentences.

We followed the approach of getting additional context by using entity pairs for querying Powerset for which we have background knowledge that the pairs are related through some relation and only retrieved the Factz that were between those entity pairs. We repeated the same experiment. However, this time we gave as input pairs of entity names for which we have evidence that the entities are related and then ran the experiment with and without semantic typing information to validate if introducing semantic typing can give us some additional advantage. We discuss the details of our experiment below.

Relations between Entity Types	Freebase Source
person - organization	PersonEmployment table
person - school	Education table
organization-organization	Acquisitions table

Table 6. Data Set with relations between different types of entities extracted from Freebase tables

Using Freebase tables we extracted datasets for relations present between three different kinds of entity pairs i.e persons and organizations (e.g. Person-join-Organization), persons and school (e.g. Person-attend-School) and Organizations and Organizations (e.g. Organization-acquire-Organization) (table 6). We used the pairs of entities (Persons - Organizations, Persons - Schools and Organizations - Organizations) to query Powerset and extracted the Factz that corresponded to those pairs. Table 7 gives an example of the predicates in the Factz found between the different types of entity pairs.

After clustering we evaluated the clusters and were expecting to get the relations between three different kinds of entity pairs namely Person - Organization, Person - School, Organization - Organization into three separate clusters. We evaluated the output clusters at different stopping thresholds but were not able to get three clusters using any threshold. Table 8 shows the clusters

found at threshold of 2. There were two possible reasons for this outcome, one reason was that we did not have enough redundancy in the data set to get meaningful clusters and secondly, “school” is a type of “organization” which could have introduced ambiguity. In order to introduce redundancy we replaced all the entity names with their types (i.e., Person, Organization, School) in the Factz and repeated the experiment with Entity Type information rather than Entity names. We evaluated the clusters at different thresholds and were able to separate the relation sets into three clusters with greater than one member. Table 9 gives the results of clustering where we got three clusters with more than one member at minimum threshold.

The clusters represented the relations present between the three different types of entity pairs i.e., person and school, organization and organization and person and organization (table 9).

Wikipedia is a very non-redundant resource and redundancy helps in getting more evidence about the similarity between relations. Other approaches (Yan et al., 2009) have used the web for getting redundant information and improving recall. In addition, there are many sentences in Wikipedia for which Powerset has no corres-

Relation	Example of Powerset Factz Predicates
Person - Organization	join, leave, found, form, start, create
Person - School	attend, enter, return to, enroll at, study at
Organization - Organization	acquire, purchase, buy, own

Table 7. Example of Predicates in Powerset Factz representing relations between different types of entity pairs

No.	Cluster Members	Semantic Types
1	enroll at, return to	Person-School
2	found, purchase, buy, acquire, create, say about, own	Organization - Organization, Person-Organization

Table 8. Results of Clustering Relations between Entity Pairs without using Semantic Typing

No.	Cluster Members	Semantic Relation
1	lead, prep at, play for, enter, study, play, graduate, transfer to, play at, enroll in, go to, remain at, enroll at, teach at, move to, attend, join, leave, teach, study at, return to, work at	Person - School
2	acquire, purchase, buy, own, say about	Organization - Organization
3	found, create	Person - Organization

Table 9. Results of Clustering Relations with Semantic Typing

ponding Factz associated (it might be due to some strong filtering heuristics). Using semantic typing helped in introducing redundancy, without which we were not able to cluster the relations between different types of entity pairs into separate clusters. Semantic Typing also helped in identifying the relations present between entities at different levels of abstraction. This can help in suggesting relations between different entity types evident in the corpus during the Ontology engineering process.

5 Conclusions

We have developed a hybrid approach for unsupervised and unrestricted relation discovery between entities using linguistic analysis via Powerset, entropy based label ranking and semantic typing information from a Knowledge base. We initially compared the accuracy of Powerset Factz with ground truth and with entropy based label ranking approach on a sample dataset and observed that the relations discovered through Powerset Factz gave higher accuracy than the entropy based approach for the sample dataset. We also developed an approach to collapse a set of relations represented in facts as a single dominating relation and introduced a hybrid approach for label selection based on relation clustering and entropy based label ranking. Our experiments showed that the relational clustering approach was able to cluster different kinds of entity pairs into different clusters. For the case where the kinds of entity pairs were at different levels of abstraction, introducing Semantic Typing information helped in introducing redundancy and also in clustering relations between different kinds of entity pairs whereas, the direct approach was not able to identify meaningful clusters. We plan to further test our approach on a greater variety of relations and on a larger scale.

References

- Dat P. T. Nguyen, Yutaka Matsuo, and Mitsuru Ishizuka. 2007. Subtree mining for relation extraction from Wikipedia. In Proc. of NAACL '07: pages 125–128.
- Dmitry Davidov, Ari Rappoport and Moshe Koppel. 2007. Fully unsupervised discovery of concept-specific relationships by Web mining. Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, pp. 232–239.
- George A. Miller , Richard Beckwith , Christiane Fellbaum , Derek Gross , Katherine Miller. 1990. Wordnet: An on-line lexical database. International Journal of Lexicography, 3(4):235-312.
- Jinxu Chen, Donghong Ji, Chew Lim Tan and Zhenyu Niu. Unsupervised Feature Selection for Relation Extraction. In Proc. of IJCNLP-2005.
- Metaweb Technologies, Freebase Data Dumps, <http://download.freebase.com/datadumps/> July, 2009
- Michele Banko , Michael J Cafarella , Stephen Soderl , Matt Broadhead , Oren Etzioni. 2008. Open information extraction from the web. Commun. ACM 51, 12 (Dec. 2008), 68-74.
- Nanda Kambhatla. 2004. Combining lexical, syntactic and semantic features with maximum entropy models. In Proceedings of the ACL 2004 on Interactive poster and demonstration sessions.
- Sanda Harabagiu, Cosmin Andrian Bejan, and Paul Morarescu. 2005. Shallow semantics for relation extraction. In Proc. of IJCAI 2005.
- Soren Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. 2007. Dbpedia: A nucleus for a web of open data. In Proc. of ISWC 2007.
- Stanley Kok and Pedro Domingos. 2008. Extracting Semantic Networks from Text Via Relational Clustering. In Proc. of the ECML-2008.
- Takaaki Hasegawa, Satoshi Sekine and Ralph Grishman. 2004. Discovering Relations among Named Entities from Large Corpora. In Proc. of ACL-04.
- Powerset. www.Powerset.com
- Yulan Yan, Naoaki Okazaki, Yutaka Matsuo, Zhenglu Yang, and Mitsuru Ishizuka. 2009. Unsupervised relation extraction by mining Wikipedia texts using information from the web. In ACL-IJCNLP '09: Volume 2, pages 1021–1029.
- Yusuke Shinyama and Satoshi Sekine. 2006. Preemptive information extraction using unrestricted relation discovery. In HLT/NAACL-2006.