Domain-Specific Relation Extraction
Using Distant Supervision Machine Learning

Abduladem Aljamel, Taha Osman and Giovanni Acampora
School of Science and Technology, Nottingham Trent University, NG11 8NS, Nottingham, U.K.

Keywords: Information Extraction, Relation Extraction, Knowledge-Base, Supervised Machine Learning, Natural Language Processing, Semantic Web.

Abstract: The increasing accessibility and availability of online data provides a valuable knowledge source for information analysis and decision-making processes. In this paper we argue that extracting information from this data is better guided by domain knowledge of the targeted use-case and investigate the integration of a knowledge-driven approach with Machine Learning techniques in order to improve the quality of the Relation Extraction process. Targeting the financial domain, we use Semantic Web Technologies to build the domain Knowledgebase, which is in turn exploited to collect distant supervision training data from semantic linked datasets such as DBPedia and Freebase. We conducted a serious of experiments that utilise the number of Machine Learning algorithms to report on the favourable implementations/configuration for successful Information Extraction for our targeted domain.

1 INTRODUCTION AND MOTIVATION

In the current digital era, an increasing amount of data is being made available online. This data can be analysed to benefit the operation of a specific domain service such as advising financial investors about a potential business risk or informing the music industry about an emerging consumer trend. The majority of that data is unstructured and constructed in natural human languages and therefore requires further processing in order to be understandable by machines and intelligently explored.

The process of extracting useful knowledge from unstructured data sources is called Information Extraction. It can be considered as a pipeline process that starts with recognising the named entities in the text, then identifying identity relation between named entities by means of co-reference resolution, and finally extracting the relation between the named entities (Cunningham 2005, Farmakiotou, et al. 2000).

A study by Cunningham (2005) shows that the complexity of the information to be extracted influences the accuracy of the Information Extraction techniques. The information complexity can vary from simple items such as people names, to complex items such as events that involve multiple participants. The authors in Cunningham (2005) conclude that the more complex the data to be extracted, the more specific must be the domain of data.

It is possible to argue therefore that specified knowledge services that require Information Extraction techniques to be able to search and extract specific knowledge directly from unstructured text should be guided by the domain knowledge that details what type of knowledge is to be obtained and for which exploration scenario. This scenario should make the Information Extraction techniques mediate between the domain text type and the requirements of various types of users.

In this paper, we adopt a knowledge-driven approach to Information Extraction that is based on comprehensive analysis of the key concepts and relations of the targeted domain in order to build a knowledgebase that informs Information Extraction processing. The knowledgebase will assist the Natural Language Processing activities as well as guide rule-based and Machine Learning techniques to infer new and interesting facts from the sourced domain data.

We argue that Semantic Web Technologies are best placed to build such knowledgebase as they are capable of organising and modelling the information into a highly structured knowledge in order to assist machines to understand information published on the

Aljamel, A., Osman, T. and Acampora, G..
Domain-Specific Relation Extraction - Using Distant Supervision Machine Learning.
In *Proceedings of the 7th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2015) - Volume 1: KDIR*, pages 92-103
ISBN: 978-989-758-158-8

Web. They describe and combine the corresponding relation between the concepts' instances from different sources and infer more information about these concepts in different contexts.

The research reported in this paper focuses on the Relation Extraction phase of Information Extraction and investigates the integration of a knowledge-driven approach with Machine Learning techniques in order to improve the quality of extracted relations from online unstructured data.

We targeted the financial domain as a use-case for implementing and evaluating our Information Extraction contribution. In recent years, analysis of published financial information has become increasingly popular to optimise business processes, inform financial trading, and reveal hidden correlations that predict relevant economic indicators (Radzimski, et al. 2012, Costantino, et al. 1997). For these financial analysis, extracting relation between domain entities is critical for the identification and correlations of the domain's key events.

The main contributions of the this work include adopting a Semantic Web based approach for constructing the Information Extraction knowledgebase, implementing and evaluating different ML classifiers for Relation Extraction, and presenting a comprehensive methodology for integrating domain knowledge with Machine Learning techniques in the Information Extraction process, which was supported by experimental analysis that presented valuable insights into a favourable configuration for training data compilation and learning algorithms setup.

The rest of this paper is organised as follows. The next section surveys the current Relation Extraction approaches and related works. The third section details the implementation of our distant supervision Relation Extraction method. Section four analyses the results of the conducted experiments. Lastly, our conclusions and plans for further works are presented in section five.

2 RELATED WORKS

There are two main approaches in Relation Extraction, rule-based and supervised Machine Learning based. The main idea of rule-based approaches is transforming the linguistic features space into lexical and syntactic patterns to be applied on natural language texts in order to extract relations. However, the relations extractors in these approaches depend on the similarity of the texts and a closed set of relations. Also, the patterns are manually crafted and a small variations in these patterns, can prevent finding appropriate relations. These patterns also are not straightforwardly applied on other domains (Garcia and Gamallo 2011, Konstantinova 2014). According to Konstantinova (2014), rule-based approaches could provide an acceptable results if the main aim is to quickly extract relations in a well linguistic defined domains. An example for this kind of approaches is a work of Akbik and Broß (2009). Their procedure uses linguistic patterns that are defined by them over the dependency grammar of sentences.

Supervised Machine Learning (ML) based approaches have been used for Information Extraction with considerable results. They could be adopted to solve problems with unstructured data; for instance, recognising named entity, classifying text and extracting relations. In general, supervised Machine Learning is about making predictions on problem solving based on information about the same problem; nonetheless, it does not require linguistics skills to be applied. An example of this kind of approaches is a study conducted by Hong (2005). He presented a Relation Extraction approach by using supervised Support Vector Machine algorithm. The classifier model created by using Automatic Content Extraction (ACE) training dataset.

The key elements of supervised Machine Learning algorithms are features vector and training datasets. The quality of these elements impacts the accuracy of algorithms' results (Daelemans and Hoste 2002, Farkas 2009, Lehmann and Völker , Jiang, et al. 2012). However, we believe that these elements could be improved if they are informed by domain knowledge.

Compilation of labelled training datasets for supervised Machine Learning classification in general and relation classification in specific is a time-consuming and cumbersome task to undertake manually; moreover, the resultant dataset is often not sufficient to solve a classification problem. Instead, there is another approach that employs existing datasets as source of supervision; it is called distant supervision or distant learning (Konstantinova 2014). Distant Supervision Machine Learning approach has emerged for training relation classification without using manually labelled data, which reduces human efforts for relation extraction (Min, et al. 2013). This approach automatically collects training examples by labelling the relations mentioned in the distant supervision sources according to the assumption that if any two entities appear in one sentence and they have a mentioned relation in at least one semantic dataset, these entities could express that relation.

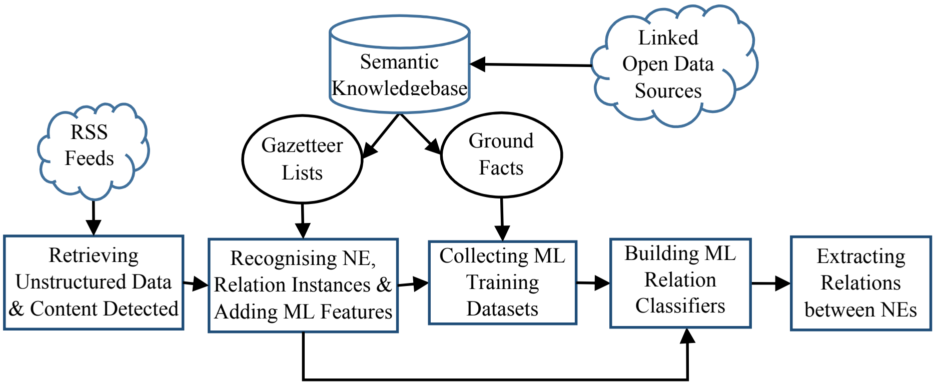
Semantic Knowledgebase
Linked Open Data Sources
RSS Feeds
Gazetteer Lists
Ground Facts
Retrieving Unstructured Data & Content Detected
Recognising NE, Relation Instances & Adding ML Features
Collecting ML Training Datasets
Building ML Relation Classifiers
Extracting Relations between NEs

Figure 1: Overview of the main tasks of our Relation Extraction methodology by using distant supervision ML approach and informed by domain knowledge.

The research work published by Mintz, et al. (2009) adopts a distant supervision approach that utilises a Freebase dataset as a distant supervision source. The Freebase's data representation was converted into binary relations. They collected 116 million instances of 7300 relations between 9 million entities. The experiments were conducted to explore each pair of entities that appears in any Freebase ground fact. Then, finding all sentences containing those entities in the unlabelled unstructured data and extracting the linguistic features to learn a relation classifier. The source of unstructured data is a dump of the full text of all Wikipedia articles. It consists of around 1.8 million articles. The ML classifier used in this research is a multi-classification logistic classifier optimised using Limited-memory Broyden–Fletcher–Goldfarb–Shanno (L-BFGS) with Gaussian regularisation. L-BFGS is an optimization algorithm uses limited amount of computer memory for parameter estimation in ML (Andrew and Gao 2007). According to Mintz, et al. (2009), the algorithm combines the advantages of supervised ML and unsupervised ML. Also, they analyse feature performance, which shows that syntactic parse features are particularly beneficial for relation classification. Their overall results showed that the distant supervision approach has the capability of extracting a high precision (67.6%) for a considerable number of relations (10000 instances of 102 relations).

Although relation classification by using distant supervision approach has been considerably studied, we believe that the aspect of integrating the knowledge-based approach in relation classification has been not extensively investigated. This work makes the following contribution to the body of the work on distant supervision relation extraction:

- We adopt Semantic Web based approach for constructing the Information Extraction Knowledgebase and utilise two semantic datasets as distant supervision sources, DBPedia and Freebase.
- We implement and evaluate three different ML classifiers for Relation Extraction, Support Vector Machine (SVM), Perceptron Algorithm Uneven Margin (PAUM) and K-Nearest Neighbour (KNN), in order to select the best ML model performance.
- We present a comprehensive methodology for integrating domain knowledge with ML techniques in the Information Extraction process.

3 RELATION EXTRACTION METHODOLOGY

The research effort reported in this paper claims that domain knowledge can significantly contribute to the activities of Information Extraction from unstructured data, and in this section we present a comprehensive, semantic-driven methodology for utilising that knowledge to improve the Machine Learning techniques employed in Relation Extraction.

Figure 1 illustrates the main tasks of the relation extraction methodology, which are summarised below and detailed in the following subsections.

1- Analysing the domain and construction of the knowledge map.
2- Retrieving unstructured data and content detection.
3- Applying Natural Language Pre-Processing tasks for:
   a. Recognising the named entities.
   b. Extracting the relation instances.
4- Adding ML features vector to relation instances.

5- Collecting the training datasets from structured datasets.
6- Building ML relation classification models.
7- Extracting Relations from the unstructured data by using the classification models.

3.1 Domain Analysis and Constructing the Knowledge Map/Ontology

In this research we analysed the use case domain and designing the knowledge map by using knowledge-driven approach. In this process we gathered knowledge in terms of what concepts to exist in a particular domain and understandable by the domain experts. Then, we translated the knowledge map into an ontology by using Semantic Web Technologies. The knowledge map assisted in analysis of our use case domain, which was key in encoding the identified entities and their interactions in a formal ontology language. When we built the ontology, we consider the reuse of publically available ontologies, particularly that in the finance domain. There are efforts to build a comprehensive taxonomies for finance domain such as a finance ontology from Fadyart (fadyart.com 2014).

For the purpose of implementation and evaluation of Relation Extraction methodology, in this paper we focused on the domain key concepts of organisations, people, and locations. The relations between the concepts are represented by the following facts:

- Person is a key person in Organisation.
- Person is a founder of Organisation.
- Organisation is an employer of Person.
- Person has location Location.
- Person has a birthplace Location.
- Person has a death place Location.
- Location is a location of Organisation.
- Person has a nationality Location.

3.2 Retrieving Unstructured Data and Content Detection

The unstructured data source is the online financial news articles. They are retrieved by using the Rich Site Summary (RSS) feeds. RSS is an XML file that links to news or other information sources in the web (Ruiz-Martínez, Valencia-García and García-Sánchez 2012). The number of documents that have been retrieved from online financial news sources is 7193 documents including the BBC, Reuters and Yahoo Finance RSS Feeds.

In addition to the actual news contents, the online news web pages consist of navigational elements, templates, and advertisements. These boilerplate texts are not related to the news contents and they may reduce the information extraction quality. As a result, they should be detected and removed properly (Kohlschütter, Fankhauser and Nejdl 2010). To achieve this task, this work employs an open source Java API library, boilerpipe from Google code (boilerpipe 2014). It provides algorithms to detect and remove the undesirable text around the main textual content of a web page. After retrieving and cleansing the online news, we store these news documents in GATE's xml format. GATE is a Natural Processing Language (NLP) tool from Cunningham, Maynard and Bontcheva (2011).

3.3 Natural Language Processing Tasks

The fundamental NLP tasks were implemented using the GATE tool. GATE is an infrastructure for developing and deploying software components that process human language texts. The aim of the NLP tasks is to produce a number of linguistic features from documents. Each document is first processed using the open source ANNIE (Nearly-New Information Extraction) system and Stanford parser, which are part of the GATE NLP tool. The NLP tasks include tokenising, sentence splitter, gazetteer lists tagging, part of speech tagging, morphological analyser, co-references resolution and dependency path tree tagging. The results of these tasks will be used for recognising the named entities and the input features vector to the ML relation classifier.

Table 1: The number of targeted named entities.

Annotation Sets Type	Entities
Organization	79545
Person	85792
Location	95134

3.3.1 Named Entity Recognition

The Named Entities are recognised by utilising ANNIE's rule-based entity recognition system, ANNIE which uses Java Annotation Patterns Engine (JAPE) rules to recognise regular expressions in annotations on documents. The ANNIE rules can be modified and extended to facilitate recognising more entities. We did extend ANNIE's JAPE rules to recognise more named entities; for instance, stock indices and stock ticker symbols. Table 1 shows the number of the extracted named entities, Person, Organization and Location.

3.3.2 Extracting Relation Instances

The baseline of relation extraction in this work is the sentence. Every entity pair for a targeted relation that appears in a sentence in unstructured data are identified and annotated as a relation instance. These pairs should be chosen to represent the relations in the domain ontology. This has been achieved by using JAPE rules and GATE Embedded Java Libraries. The number of sentences and relation instances of the targeted relations in this work are shown in Table 2.

Table 2: The sentences and relation instances number (RI=Relation Instances, Per=Person, Org=Organization and Loc=Location).

Annotation Type	Number
Sentences	316504
RI of Per-Org pairs	32304
RI of Per-Loc pairs	38425
RI of Loc-Org pairs	28891

3.4 Adding Ml Features Vector to Relation Instances

ML classification tasks require assigning features vector to a finite set of classes. Features represent any distinctive aspects, qualities or characteristics of classes. They may be symbolic such as Part Of Speech of an entity, or numeric such as the number of words between two entities. The quality of features vector is one of the key factors of any classification technique performance (Han, Kamber and Pei 2011).

Table 3: Machine Learning Features Vector list.

Features Category	Description
Lexical features	POS of words between entity pairs.
	General POS of words between entity pairs.
	POS of three words before the first entity.
	POS of three words after the second entity.
Syntactic Features	The words' strings of collapsed typed dependency path between entity pairs.
	The kinds of collapsed typed dependency path between entity pairs.
	The whole collapsed typed dependency path of the entity pairs' sentences.
	Direct collapsed typed dependency path between entity pairs.
Named Entity Features	The size of the first entity.
	The size of the second entity.
	The order of the entities.
	The distance between the two entities.
	Token string of the first entity.
	Token string of the second entity.

Khan and Baig (2015) argue that the sufficient domain knowledge could assist in selecting the features vector as input to the classification algorithms. We expanded on the feature set suggested by Mintz, et al. (2009) to provide a more comprehensive set of syntactic features; for instance, the typed dependency path between words contains information which could be used to reflect the relation environment in the sentence. Each feature attempts to describe the characteristics of the relation between two entities in a sentence. They could be categorised into three types, lexical features, syntactic features and named entity features (see Table 3). These features are extracted by using JAPE rules and GATE Embedded and added to every relation instances in the unstructured data.

Below is a JAPE code example:

```
Phase: instances
Input: Sentence
Options: control = all
Rule: pair
({Sentence}):s
-->
{ [In this part, GATE Embedded JAVA
libraries can be coded to extract and
annotate information from the annotation
patterns in the upper part] }
```

3.5 Collecting Training Datasets from Online Structured Datasets

The distant supervision approach in this work, employs existing semantic datasets, DBPedia and Freebase, as a distant supervision sources for ML relation classification. DBpedia contains more than 4.5 million entities and more than 3 billion RDF triple for a diversity of language. Freebase dataset contains approximately 47.5 million topic and 2.9 billion facts in English language.

The training datasets were built by retrieving the relations between any two entities in a single sentence in the unstructured document and mentioned in Freebase or DBPedia as ground facts. These relations are assumed to be a class instance or true positive in the training datasets. The mentioned relations in the semantic datasets were extracted by using JENA's SPARQL engine, which facilitates the retrieval of the relations on RDF semantic format (Harris, Seaborne and Prud'hommeaux 2013). To illustrate this task, we use the following sentence example from the unstructured data corpus which is used in this work:

```
"Yesterday Twitter's boss Dick Costolo
said he was "ashamed" at how the site
had dealt with abusive online trolls."
```

The sentence contains the following relation instance:

```
"Twitter's boss Dick Costolo"
```

The relation instance contains two entities, Person entity "Dick Costolo" and Organization entity "Twitter".

These two entities' names are used to query the semantic datasets to find if they have any mentioned relation. The SPARQL query for this example and its result are shown below.

```
PREFIX  rdfs:
<http://www.w3.org/2000/01/rdf-schema#>
SELECT DISTINCT  (str(?lbl) AS ?result)
WHERE {
        { ?entity1 ?rel ?entity2 .
          ?entity1 rdfs:label "Dick
Costolo"@en .
          ?entity2 rdfs:label
"Twitter"@en
        }
        UNION
        { ?entity2 ?rel ?entity1 .
          ?entity1 rdfs:label
"Twitter"@en .
          ?entity2 rdfs:label "Dick
Costolo"@en
        }
        ?rel rdfs:label ?lbl
        FILTER ( lang(?lbl) = "en" )
      }
```

```
-------------
| result    |
=============
|"employer" |
-------------
```

This result means that the relation mentioned in the semantic dataset in the form of RDF triple is as follows:

```
"Twitter employer Dick Costolo"
```

This relation is mapped into a relation in domain's ontology as:

```
"Twitter employerOf Dick Costolo"
```

Then, the relation is assumed as a class instance or True Positive in the Organization-Location training dataset.

Table 4 shows the three training datasets that were collected by using distant supervision approach adopted by this work. The first training dataset represents relations between Person and Organization entities with three relation classes. The second training dataset represents relations between Person and Location entities with four relation classes. The third and last training dataset represents relations between Location and Organization entities with one relation class. In addition to the number of the documents in each training dataset, Table 4 also presents the total number of all mentioned relations for all classes in the training datasets, the total number of mentioned relations for each class, and the total number of relation instances including those that are not mentioned in DPBedia and Freebase and present in the documents of the training datasets.

Table 4: The summary of the collected training datasets (RI=all Relation Instances, MR= Mentioned Relations, Doc=Documents, Per=Person, Org=Organization and Loc=Location).

Entity Pairs	**Doc**	**RI**	**MR**	**Relation Types**	
Per-Org	192	4213	204	founderOf	38
				keyPersonIn	107
				employerOf	59
Per-Loc	671	11152	896	hasPlace	221
				birthplace	233
				hasNationality	415
				deathPlace	27
Loc-Org	581	6217	299	locatedIn	299

3.6 Building Machine Learning Classification Models

In this section, we detail the ML algorithms that were trained by using the training datasets and features vector in order to build the models for relation classification. These models will be evaluated to establish the suitable ML algorithm for relation extraction. The next section describes in details the ML classifiers adopted in this work.

The three supervised ML classifiers applied for relations classification by using distant supervision are, Support Vector Machine (SVM), Perceptron Algorithm Uneven Margin (PAUM) and K-Nearest Neighbour (KNN). The works of Panchenko, et al. (2012), Hmeidi, Hawashin and El-Qawasmeh (2008), Li, Bontcheva and Cunningham (2009), Li, et al. (2005), and Witten and Frank (2005) reveal that SVM, PAUM and KNN are used in Information Extraction tasks with adequate results.

SVM is a supervised ML algorithm that has an advanced performance for a diversity of classification tasks including Information Extraction specifically in small training datasets. One of the striking features of SVM is that it has a robust justification for avoiding over fitting (Cunningham, Maynard and Bontcheva 2011, Wang, et al. 2006). SVM is an optimal classifier, which means that it learns a classification hyperplane in the features space with the maximal margin to all training instances (Li, Bontcheva and Cunningham 2009). This work uses the GATE implementation, which is based on Java version of the SVM package LibSVM with exception that the GATE implements the uneven margins SVM algorithm that described in the work of Li, Bontcheva and Cunningham (2009). The most important parameters of this implementation are SVM cost (C, the Cost associated with allowing training errors, soft margin) and the uneven margins (τ or tau, setting the value of uneven margins parameter of the SVM) (Li, Bontcheva and Cunningham 2009, Li and Shawe-Taylor 2003). There are several kernel functions types can be used in SVM algorithm such as linear, polynomial, radial and sigmoid. Li, Bontcheva and Cunningham (2009) mentioned in their research that linear kernel is much more computationally efficient than other complicated kernel functions and tends to obtain similar performance; therefore, in this research we used linear kernel. Also, the values of C and tau parameters which are used in this research are 1 and 0.8 respectively.

PAUM is a simple and effective learning algorithm especially for large training datasets. It has been successfully used for document classification and information extraction. For a binary classification problem, it checks each instances in the training dataset by predicting their labels. If the prediction is correct, the instance is passed; otherwise, it is used to correct the model. The algorithm stops when the model classifies all training instances correctly. The utilised GATE implementation of the PAUM algorithm proposes two margin parameters, positive and negative. These two margin parameters allow the PAUM to handle imbalanced datasets better. Also, GATE implementation proposes the modification of the bias term parameter (optB). The values of negative and positive margins and optB parameters which are used in this work are 1, 50 and 0.3 respectively (Li, et al. 2005, Cunningham, Maynard and Bontcheva 2011).

KNN is a simple and often its accuracy is enhanced when the number of features is small. It is an instance-based classification, which means that each new instance is compared with K nearest neighbour instances by using a distance metric. The class that has the majority of instances of the closest K neighbours is assigned to the new instance. KNN algorithm shows superior results in classifying documents. However, it is a lazy learning algorithm because it depends only on statistics. KNN implementation used in this work has only one parameter, K, which can be tuned heuristically in order to find the best algorithm's performance. The implementation of this algorithm is provided by GATE, which is based on the open source ML package WEKA (Hmeidi, Hawashin and El-Qawasmeh 2008, Witten and Frank 2005). The value of K which is used in this work is 1.

The algorithms above can implement both binary and multi-class classifiers; however, the implementation of multi-classification ML algorithms is more complicated than binary classification. As a result, when using an effective binary classifier, the scheme of converting the multi-classification to multiple binary classifications by using a simple "one-vs-others" or "one-vs-another" methods is preferred over other complex methods such as complex error-correcting coding method. The "one-vs-others" method converts N class classification model (N>2) into N binary classification models. In every binary classification model, the positive instances are belonging to a specific class and the negative instances are belonging to all other classes. In contrast, "one-vs-another" method converts N class classification model (N>2) into N(N−1)/2 binary classification models of class pairs. In every binary classification model, the positive instances are belonging to one class in the pair and negative instances are belonging to the other class in the same pair (Li, Bontcheva and Cunningham 2009). Therefore, this paper only considers the "one-vs-others" method in transforming multi-classifier into multiple binary classifier because it requires less number of models.

The section above presented our methodology for implementing knowledge-driven Relation Extraction that include constructing the knowledge map, Named Entity Recognition, relation instances tagging, collecting training datasets, retrieving features set and collecting dataset. In following section we are going to evaluate the Machine Learning algorithms with the features vector and training datasets.

4 RESULTS EVALUATION AND DISCUSSION

The ML relation classification model has been created by using the training datasets that were collected by adopting the distant supervision approach with the features vector. The models should be evaluated before applying them to extract relations from unstructured data. We evaluate the ML models to get the optimum results by configuring training datasets.

There are two commonly used evaluation methods for ML algorithms, K-fold cross-validation and holdout test. In K-fold cross-validation, the corpus is split into K equal size partitions of documents. The evaluation run is repeated K times, folds. Each partition is used as test dataset and all the remaining partitions as a training dataset for all K folds. The overall Recall, Precision and F1-measure result of this method is the average of the all folds results. In contrast, in holdout test, a number of documents in corpus are randomly selected according to a specified ratio. These documents are assumed as testing dataset and all other documents as training dataset (Cunningham, Maynard and Bontcheva 2011). In this work, the experiments have been conducted on the training datasets by using cross validation K-Fold with K=10. This method guarantee that all documents will participate in the training and testing datasets.

According to Witten, Frank and Hall (2005), there is more than one method to compute the evaluation of ML algorithms performance. These methods depend on the target domain. For instance, the marketing domain uses lift chart by plotting True Positive rate versus training subset size, the communication domain uses Receiver Operator Characteristic (ROC) curve by plotting True Positive rate versus False Positive rate and the Information Retrieval domain uses Recall versus Precision curve. This research computes the evaluation results of ML models in relation classification by drawing the relation between recall and precision in terms of threshold probability classification confidence values.

A series of experiments have been conducted in this research in order to improve the accuracy of ML models and choose between ML algorithms, SVM, PAUM and KNN.

4.1 Machine Learning Accuracy Experiments

The first set of experiments attempts to alleviate the classes imbalance in terms of True Positive and True Negative numbers and to speed up ML processing by reducing or removing the relation instances in the documents that are not mentioned in the distant supervision sources. Some of these instances are not tagged as True Positive solely because they are not populated onto Freebase or DBPedia datasets. Table 4 shows the total number of the relation instances and total number of mentioned relation instances in each training dataset.

The ML algorithms in this research transforms multi-classification into multiple binary-classification technique using "one-vs-others" method. This method assumes the instances of one relation class as True Positive and the instances of the other relation classes as True Negative in each binary classification for each relation class. The results of this method are the predictions of the highest confidence scored instances. In this experiments' set we measure the impact of reducing the number of not mentioned relation instances on models accuracy by reducing them gradually and calculating Precision, Recall and F1-measure every time until reaching better ML model accuracy.

Table 5 compares between the F1-measure values of SVM, PAUM and KNN models when using the training datasets with different numbers of relation instances. The Person-Organisation pair contains four classes, the Person-Location pair contains three classes and Location-Organization pair contains one class. Figures 2, 3 and 4 presents Precision-Recall charts to compare between SVM models performance when reducing the number of not mentioned relation instances in the training datasets. These figures show there is a clear trend of an increase in precision and recall of ML models in the lowest values of the number of the not mentioned relation instances.

Table 5: It shows the impact of reducing the number of Not Mentioned Relation Instances (NMRI) in ML models accuracy. (Per=Person, Org=Organisation, and Loc=Location).

Pair	Per-Org		Per-Loc		Loc-Org	
NMRI	**0**	**51**	**0**	**224**	**12**	**74**
SVM	***0.80***	0.78	***0.71***	0.66	***0.90***	0.83
PAUM	**0.77**	0.75	**0.68**	0.64	**0.90**	0.83
KNN	**0.77**	0.72	**0.65**	0.60	**0.89**	0.81

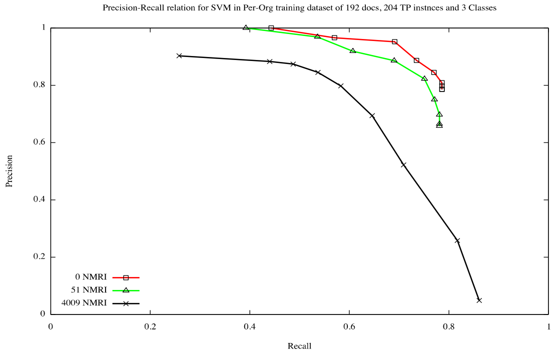

Figure 2: SVM model accuracy in terms of number of not mentioned relation instances in Person Organization pair training dataset. (NMRI=Not Mentioned Relation Instances).

Table 5 and Figures 2, 3 and 4, indicate that the best precision, recall and F1-measure of ML models is achieved when removing or reducing the number of relation instances that are not mentioned as a ground fact in DBPedia or Freebase from the training datasets, which can be explained by the fact that these instances are considered to be an additional class instances that disrupt the balance between True Positives and True Negatives in the training datasets. The work of Mintz, et. al (2009) indicates that these negative instances have a minor effect on the performance of the classifier. However, they used a pure multi-classification classifier; in contrast to our implementation that converts multi-classification into multiple binary classification.

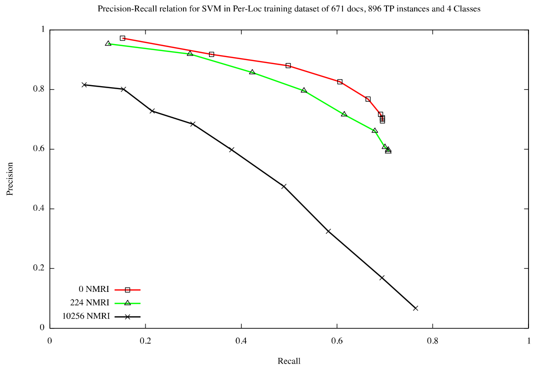

Figure 3: SVM model accuracy in terms of number of not mentioned relation instances in Person Organization pair training dataset. (NMRI=Not Mentioned Relation Instances).

Table 6: The summary of the predicted relations (PR=Predicted Relation, Per=Person, Org=Organization, and Loc=Location).

Entity Pairs	Relation Types	PR
Per-Org	founderOf	229
	keyPersonIn	4718
	employerOf	3062
Per-Loc	hasPlace	10274
	birthplace	6294
	hasNationality	11822
	deathPlace	53
Loc-Org	locatedIn	5004

4.2 Selecting Optimum Machine Learning Classifier

The second set of experiments aims to identify the ML classifier amongst SVM, PAUM and KNN. The results of these experiments are illustrated in Table 5 and Figures 5, 6 and 7. The results indicate that SVM is better than the PAUM and KNN algorithms in terms of F1-measure also in terms of precision-recall relation. These results agree with the findings of other studies; for example, in a study of Li, et al. (2005) found that SVM may perform better than PAUM in small training datasets and they have a close performance in large training datasets. Also, the work of Hmeidi, Hawashin and El-Qawasmeh (2008) reveal that SVM has better F1-measure results than KNN.

Our analysis assert that SVM relation classifier exhibit more accurate results, which we attribute to two reasons. Firstly, the training datasets of this work are relatively small because of the characteristics of the financial and economic domain that it is not common in DBPedia and Freebase comparing to other public domains such as music and movies domains. Secondly, it has a superior performance in small training datasets. Therefore, this research has adopted SVM algorithm to create the relation classification model.

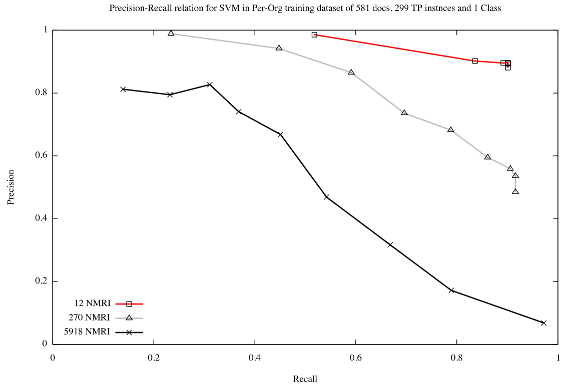

Figure 4: SVM model accuracy in terms of number of not mentioned relation instances in Location Organization pair training dataset. (NMRI=Not Mentioned Relation Instances).

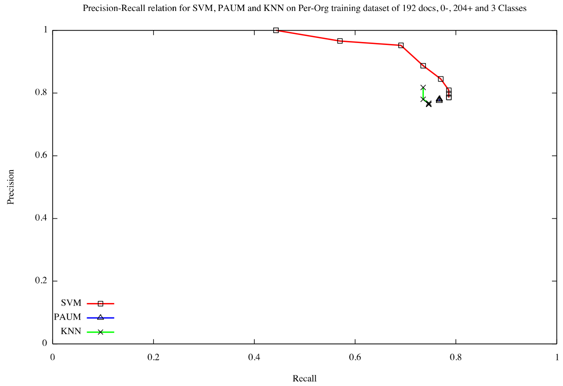

Figure 5: Comparison between SVM, PAUM and KNN models in Person Organization pair training dataset.

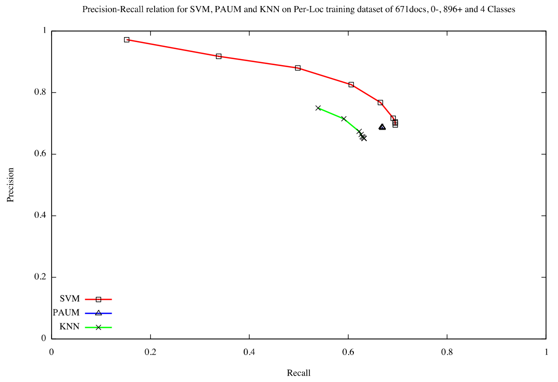

Figure 6: Comparison between SVM, PAUM and KNN models in Person Location pair training dataset.

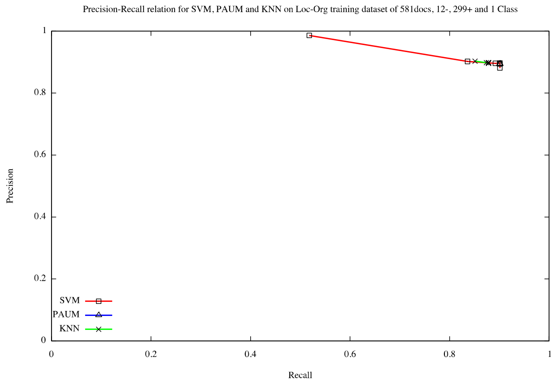

Figure 7: Comparison between SVM, PAUM and KNN models in Location Organization pair training dataset.

4.3 Relation Classification

The relation classification model was created using SVM algorithm. This model takes the unlabelled relation instances in the unstructured documents with the features vector as an input. Then, it returns a relation names for those relation instances with a confidence score based on the probability of the correctness of entity pairs relation. The confidence score could be used to rank the relations to be used to generate a list of the most confident relations (Mintz, et al. 2009). Table 6 shows the number of predicted relations between the targeted named entities.

It can be seen from the results of this work that Freebase and DBPedia provide a sufficient ground facts for the Organization, Location and Person concepts interrelations. Nonetheless, there are other relations between other entity pairs that require more investigation; such as, stock markets and organizations entities interrelations. While the focus of this research is on the financial and economic domain use case, the structured knowledge sources have a rich set of ground facts for a variety of domains such as, sports, entertainment, politics and others, which confirms that our approach is reusable and applicable for a variety of domains.

5 CONCLUSIONS AND FURTHER WORK

Online data can be exploited to inform data analytics and decision support systems for a variety of applications such as those belonging to the financial services domain. For this class of applications, extracting relations between key domain concepts is critical for the identification and correlation of the domain's key events.

In this paper, we argue that extracting information for a specific domain should be guided by the domain knowledge that describes what type of knowledge is to be obtained and for which exploitation scenario, and subsequently present a comprehensive methodology for integrating domain knowledge with machine learning techniques in order to improve the information extraction process. We model the domain's key concepts and its interaction with the beneficiary users using Semantic Web technologies, which provides for organising information database into a highly structured knowledgebase and also allows reasoning on the sourced data to infer new interesting facts. Moreover, the semantic knowledgebase allows us to seamlessly source data from Linked Datasets such as Freebase and DBpedia that also use Semantic technology to store domain-relevant ground facts.

We address the difficulties of training datasets provision for supervised machine learning classification by adopting a distant supervision machine learning. Our approach automatically collects training examples by labelling the relations mentioned in distant supervision sources hosted in semantic linked datasets.

A series of experiments have been conducted in this research in order to select the algorithm that promoted better relation extraction accuracy, and concluded that SVM outperforms PAUM and KNN algorithms in terms of F1-measure and the precision-recall relation. Another important finding of our experimental work is that the relation extraction accuracy is improved when reducing the number of relation instances that are not mentioned as a ground fact in DBPedia or Freebase, which can be explained by the fact that these instances are considered to be an additional class instances that disrupt the balance between True Positives and True Negatives in the training datasets.

The distant supervision training datasets sources (DBPedia and Freebase) have a rich set of ground facts for diverse domains such as, sports, entertainment, politics and others, which confirms that our approach is reusable and applicable for a variety of domains.

Our plans for further work include investigating the use of evolutionary algorithms for feature selection and implementing semantic rules that infer implicit facts from the knowledgebase to support financial decision making.

REFERENCES

Akbik, A., and Broß, J., 2009. Wanderlust: Extracting semantic relations from natural language text using dependency grammar patterns. *In: WWW Workshop*.

Andrew, G., and Gao, J., 2007. Scalable training of L 1-regularized log-linear models. *In: Proceedings of the 24th international conference on Machine learning,* ACM, pp. 33-40.

boilerpipe, 2014. *boilerpipe* [online]. Google. Available at: https://code.google.com/p/boilerpipe [Accessed 5/20 2014].

Costantino, M., Morgan, R.G., Collingham, R.J. and Carigliano, R., 1997. Natural language processing and information extraction: Qualitative analysis of financial news articles. *In: Computational Intelligence for Financial Engineering (CIFEr), 1997., Proceedings of the IEEE/IAFE 1997,* IEEE, pp. 116-122.

Cunningham, H., 2005. Information extraction, automatic. *Encyclopedia of Language and Linguistics*, 665-677.

Cunningham, H., Maynard, D. and Bontcheva, K., 2011. *Text processing with gate.* Gateway Press CA.

Daelemans, W., and Hoste, V., 2002. Evaluation of machine learning methods for natural language processing tasks. *In: 3rd International conference on Language Resources and Evaluation (LREC 2002),* European Language Resources Association (ELRA).

fadyart.com, 2014. *Finance Ontology*[online]. fadyart.com. Available at: http://fadyart.com [Accessed 4/30 2014].

Farkas, R., 2009. Machine learning techniques for applied information extraction.

Farmakiotou, D., Karkaletsis, V., Koutsias, J., Sigletos, G., Spyropoulos, C.D. and Stamatopoulos, P., 2000. Rule-based named entity recognition for Greek financial texts. *In: Proceedings of the Workshop on Computational lexicography and Multimedia Dictionaries (COMLEX 2000),* Citeseer, pp. 75-78.

Garcia, M., and Gamallo, P., 2011. A Weakly-Supervised Rule-Based Approach for Relation Extraction. *In: XIV Conference of the Spanish Association for Artificial Intelligence (CAEPIA 2011),* pp. 07-2011.

Han, J., Kamber, M. and Pei, J., 2011. *Data mining: concepts and techniques: concepts and techniques.* Elsevier.

Harris, S., Seaborne, A. and Prud'hommeaux, E., 2013. SPARQL 1.1 query language. *W3C Recommendation,* 21.

Hmeidi, I., Hawashin, B. and El-Qawasmeh, E., 2008. Performance of KNN and SVM classifiers on full word Arabic articles. *Advanced Engineering Informatics,* 22 (1), 106-111.

Hong, G., 2005. Relation extraction using support vector machine. *In:* Relation extraction using support vector machine. *Natural Language Processing–IJCNLP 2005.* Springer, 2005, pp. 366-377.

Jiang, X., Huang, Y., Nickel, M. and Tresp, V., 2012. Combining information extraction, deductive reasoning and machine learning for relation prediction. *In:* Combining information extraction, deductive reasoning and machine learning for relation prediction. *The Semantic Web: Research and Applications.* Springer, 2012, pp. 164-178.

Khan, A., and Baig, A.R., 2015. Multi-Objective Feature Subset Selection using Non-dominated Sorting Genetic Algorithm. *Journal of Applied Research and Technology,* 13 (1), 145-159.

Kohlschütter, C., Fankhauser, P. and Nejdl, W., 2010. Boilerplate detection using shallow text features. *In: Proceedings of the third ACM international conference on Web search and data mining,* ACM, pp. 441-450.

Konstantinova, N., 2014. Review of Relation Extraction Methods: What Is New Out There? *In:* Review of Relation Extraction Methods: What Is New Out There? *Analysis of Images, Social Networks and Texts.* Springer, 2014, pp. 15-28.

Lehmann, J., and Völker, J., Perspectives on Ontology Learning.

Li, Y., Bontcheva, K. and Cunningham, H., 2009. Adapting SVM for data sparseness and imbalance: a case study in information extraction. *Natural Language Engineering,* 15 (02), 241-271.

Li, Y., Miao, C., Bontcheva, K. and Cunningham, H., 2005. Perceptron Learning for Chinese Word Segmentation. *In: Proceedings of Fourth SIGHAN Workshop on Chinese Language Processing (Sighan-05),* pp. 154-157.

Li, Y., and Shawe-Taylor, J., 2003. The SVM with uneven margins and Chinese document categorization. *In: Proceedings of The 17th Pacific Asia Conference on Language, Information and Computation (PACLIC17),* pp. 216-227.

Min, B., Grishman, R., Wan, L., Wang, C. and Gondek, D., 2013. Distant Supervision for Relation Extraction with an Incomplete Knowledge Base. *In: HLT-NAACL,* pp. 777-782.

Mintz, M., Bills, S., Snow, R. and Jurafsky, D., 2009. Distant supervision for relation extraction without labeled data. *In: Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2,* Association for Computational Linguistics, pp. 1003-1011.

Panchenko, A., Adeykin, S., Romanov, P. and Romanov, A., 2012. Extraction of semantic relations between concepts with knn algorithms on wikipedia. *In: Concept Discovery in Unstructured Data Workshop (CDUD) of International Conference On Formal Concept Analysis, Belgium,* Citeseer, pp. 78-88.

Radzimski, M., Sánchez-Cervantes, J.L., Rodríguez-González, A., Gómez-Berbís, J.M. and García-Crespo, Á, 2012. FLORA–Publishing Unstructured Financial Information in the Linked Open Data Cloud. *In: International Workshop on Finance and Economics on the Semantic Web (FEOSW 2012),* pp. 27-28.

Ruiz-Martínez, J.M., Valencia-García, R. and García-Sánchez, F., 2012. Semantic-Based Sentiment analysis in financial news. *In: Proceedings of the 1st International Workshop on Finance and Economics on the Semantic Web,* pp. 38-51.

Wang, T., Li, Y., Bontcheva, K., Cunningham, H. and Wang, J., 2006. *Automatic extraction of hierarchical relations from text.* Springer.

Witten, I.H., and Frank, E., 2005. *Data Mining: Practical machine learning tools and techniques.* Morgan Kaufmann.