Towards a stepwise method for unifying and reconciling corporate names in public contracts metadata. The CORFU technique.

Jose María Álvarez[1], Michail Vafolopoulos[2], and José Emilio Labra[3]

[1] South East European Research Center,
54622, Thessaloniki, Greece
jmalvarez@seerc.org
[2] Multimedia Technology Laboratory, National Technical University of Athens,
15773, Athens, Greece.
vafopoulos@medialab.ntua.gr,
[3] WESO Research Group, Department of Computer Science, University of Oviedo,
33007, Oviedo, Spain.
labra@uniovi.es.es

Abstract. The present paper introduces a technique to deal with coporate names heterogeneities in the context of public procurement metadata. Public bodies are currently facing a big challenge trying to improve both the performance and the transparency of administrative processes. The e-Government and Open Linked Data initiatives have emerged as efforts to tackle existing interoperability and integration issues among ICT-based systems but the creation of a real transparent environment requires much more than the simple publication of data and information in specific open formats; data and information quality is the next major step in the pubic sector. More specifically in the e-Procurement domain there is a vast amount of valuable metadata that is already available via the Internet protocols and formats and can be used for the creation of new added-value services. Nevertheless the simple extraction of statistics or creation of reports can imply extra tasks with regards to clean, prepare and reconcile data. On the other hand, transparency has become a major objective in public administractions and, in the case of public procurement, one of the most interesting services lies in tracking rewarded contracts (mainly type, location, and supplier). Although it seems a basic kind of reporting service the truth is that its generation can turn into a complex task due to a lack of standardization in supplier names or the use of different descriptors for the type of contract. In this paper, a stepwise method based on natural language processing and semantics to address the unfication of corporate names is defined and implemented. Moreover a research study to evaluate the precision and recall of the proposed technique, using as use case the public dataset of rewarded public contracts in Australia during the period 2004-2012, is also presented. Finally some discussion, conclusions and future work are also outlined.

1 Introduction

Public bodies are continuously publishing procurement opportunities in which valuable metadata is available. Depending on the stage of the process new data arises such as the supplier name that has been rewarded with the public contract. In this context the extraction of statistics on how many contracts have been rewarded to the same company is a relevant indicator to evaluate the transparency of the whole process. Although companies that want to tender for a public contract must be officially registered and have an unique identification number, the truth is that in most of rewarded contracts the supplier is only identified by a name or a string literal typed by a civil-servant. In this sense there is not usually a connection between the official company registry and the process of rewarding contracts implying different naming problems and data inconsistencies that are spread to next stages preventing future activities such as reporting.

In the case of the type of contract and location, there are already standardized [28] product scheme classifications such as the Common Procurement Vocabulary (2003 and 2008), the Combined Nomenclature (2012), the Central Product Classificationby in the European Union, the International Standard Industrial Classification of All Economic Activities (Rev. 4) in the United Nations or the North American Industry Classification System (2007 and 2012) in the Government of United States that are currently used with different objectives such as statistics, tagging or information retrieval. Geolocated information can be also found in different common datasets and nomenclatures such as the Nomenclature of territorial units for statistics in the European Union, the Geonames dataset [4], the GeoLinkedData initiative [18] or the traditional list of countries and ISO-codes.

However corporate, organization, firm, company or institution names (hereafter these names will be used to refer to the same entity) and structure is not yet standardized at global scope and only some classifications of economic activities or company identifiers can be found such as the TARIC (On-line customs tariff database) database. Thus the simple task of grouping contracts by a supplier is not a mere process of searching by the same literal. Technical issues such as hyphenation, use of abbreviations or acronyms an transliteration are common problems that must be addressed in order to provide a final corporate name. Existing works in the field of Name Entity Recognition [23] (NER) or name entity disambiguation [29,12] have already addressed these issues. Nevertheless the problem that is being tackled in these approaches lies in the identification of organization names in a raw text while in the e-Procurement sector the string literal identifying a supplier is already known.

In the particular case of the Australian e-Procurement domain, the supplier name seems to be introduced by typing a string literal without any assistance or auto-complete method. Obviously a variety of errors and variants for the same company, see Table 3 in the Appendix I, can be found such as misspelling errors [24,15], name and acronym mismatches [31,26] or context-aware data that

[4] `http://www.geonames.org/`

is already known when the dataset is processed, e.g. country or year. Furthermore it is also well-known that a large company can be divided into several divisions or departments but from a statistical point of view grouping data by a supplier name should take into account all rewarded contracts regardless the structure of the company.

On the other hand the application of semantic technologies and the Linking Open Data initiative (hereafter LOD) [2,9] in several fields like e-Government (e.g. the Open Government Data effort) tries to improve the knowledge about a specific area providing common data models and formats to share information and data between agents. More specifically, in the European e-Procurement context [5] there is an increasing commitment to boost the use of electronic communications and transactions processing by government institutions and other public sector organizations in order to provide added-value services with special focus on SMEs. More specifically the LOD initiative seeks for creating a public and open data repository in which one the principles of this initiative that lies in the unique identification or resources through URIs can become real. Thus entity reconciliation techniques [1,19] coming from the ontology mapping and alignment areas or algorithms based on Natural Language Processing (hereafter NLP) have been designed to link similar resources already available in different vocabularies, datasets or databases such as DBPedia or Freebase. Nevertheless the issue of unifying supplier names as a human would do faces new problems that have been tackled in other research works [6] to extract statistics of performance in bibliographic databases. The main objective is not just a mere reconciliation process to link to existing resources but to create a unique name or link (n string literals $\rightarrow$ 1 company $\rightarrow$ 1 URI). For instance in the case of the ongoing example the string literals "Oracle" and "Oracle University" could be respectively aligned to the entity <Oracle_Corporation> and <Oracle_University> but the problem of grouping by a unique (Big) name, identifier or resource still remains. That is why a context-aware method based on NLP techniques combined with semantics has been designed, customized and implemented trying to exploit the naming convention of a specific dataset with the aim of grouping n string literals $\rightarrow$ 1 company and, thus, easing the next natural process of entity reconciliation.

The remainder of this paper is structured as follows. Section 2 a literature review is presented. Afterwards next section outlines main mismatches in corporate names. Section 4 presents the CORFU approach to unify corporate names. The evaluation section exposes and discusses the experimentation carried out to test the presented approach using as a dataset the rewarded contracts of Australia in the period 2004-2012. Finally conclusions summarizes the main outcomes of this work and some open issues are also presented as future work.

2 Related Work

According to the previous section, some related work can be found in the areas this work covers:

- Natural Language Processing and Computational Linguistics. In these research areas common works dealing with the aforementioned data heterogeneities such as misspelling errors [24,15] and name/acronym mismatches [31,26], in the lexical, syntactic and semantic levels can be found. These approaches can be applied to solve general problems and usually follow a traditional approach of text normalization, lexical analysis, pos-tagging word according to a grammar and semantic analysis to filter or provide some kind of service such as information/knowledge extraction, reporting, sentiment analysis or opinion mining. Well-established APIs such as NLTK [17] for Python, Lingpipe [3], OpenNLP [14] or Gate [4] for Java, WEKA [27] (a data mining library with NLP capabilities), the Apache Lucene and Solr [25] search engines provide the proper building blocks to build natural-language based applications. Recent times have seen how the analysis of social networks such as Twitter [16,8], the extraction of clinical terms [30] for electronic health records, the creation of bibliometrics [6,21] or the identification of gene names [13,7] to name a few have tackled the problem of entity recognition and extraction from raw sources. Other supervised techniques [22] have also be used to train data mining-based algorithms with the aim of creation classifiers.
- Semantic Web. More specifically in the LOD initiative [2] the use of entity reconciliation techniques to uniquely identify resources is being currently explored. Thus an entity reconciliation process can be briefly defined as the method for looking and mapping [10,11] two different concepts or entities under a certain threshold. There are a lot of works presenting solutions about concept mapping, entity reconciliation, etc. most of them are focused on the previous NLP techniques [19,1] (if two concepts have similar literal descriptions then they should be similar) and others (ontology-based) that also exploit the semantic information (hierarchy, number and type of relations) to establish a potential mapping (if two concepts share similar properties and similar super classes then these concepts should be similar). Apart from that there are also machine learning techniques to deal with these mismatches in descriptions using statistical approaches. Recent times, this process has been widely studied and applied to the field of linking entities in the LOD realm, for instance using the DBPedia [20]. Although there is no way of automatically creating a mapping with a 100% of confidence (without human validation) a mapping under a certain percentage of confidence can be enough for most of user-based services such as visualization. However, in case of using these techniques as previous step of a reasoning or a formal verification process this ambiguity can lead to infer incorrect facts and must be avoided without a previous human validation.

  On the other hand the use of semantics is also being applied to model organizational structures. In this case the notion of *corporate* is presented in several vocabularies and ontologies as Dave Reynolds (Epimorphics Ltd) reports [5]. Currently the main effort is focused in the designed of the Organizations Vo-

[5] `http://www.epimorphics.com/web/wiki/organization-ontology-survey`

cabulary (a W3C Working Draft) in which the structure and relationships of companies are being modeled. This proposal is especially relevant in the next aspects: 1) to unify existing models to provide a common specification; 2) to apply semantic web technologies and the Linked Data approach to enrich and publish the relevant corporate information; 3) to provide access to the information via standard protocols and 4)to offer new services that can exploit this information to trace the evolution and behavior of the organization over time.

- Corporate Databases. Although corporate information such as identifier, name, economic activity, contact person, address or financial status is usually publicly available in the official government registries the access to this valuable information can be tedious due to different formats, query languages, etc. That is why other companies have emerged trying to index and exploit these public repositories; selling reporting services that contain an aggregated version of the corporate information. Taking as an example the Spanish realm, the Spanish Chambers of Commerce [6], Empresia.es [7] or Axesor.es [8] manage a database of companies and individual entrepreneurs. This situation can be also transpose to the international scope, for instance Forbes keeps a list of the most representative companies in different sectors. The underlying problems lies in the lack of unique identification, same company data in more than a source, name standardization, etc. and, as a consequence, difficulty of tracking company activity. In order to tackled these problems some initiatives applying the LOD principles such as the Orgpedia [9] in United States or "The Open Database Of The Corporate World" [10] have scrapped and published the information of companies creating a large database containing ($54,080,317$ of companies in May 2012) with high-valuable information like the company identifier. Apart from that, reconciliation services have also been provided but the problem of mapping (n string literals $\rightarrow$ 1 company $\rightarrow$ 1 URI, as a human would do and the previous section has presented) still remains. Finally public web sites and major social networks such as Google Places, Google Maps, Foursquare, Linkedin Companies or Facebook provide APIs and information managed by the own companies that is supposed to be specially relevant to enrich existing corporate data once a company is uniquely identified.

3 The CORFU technique

According to [6,21] institutional name variations can be classified into two different groups: 1) Non-acceptable variations (affect to the meaning) due to misspelling or translation errors and 2) acceptable variations (do not affect to the

[6] `http://www.camerdata.es/php/eng/fichero\_empresas.php`
[7] `http://empresia.es`
[8] `http://www.axexor.es`
[9] `http://tw.rpi.edu/orgpedia/`
[10] `http://opencorporates.com/`

meaning) that correspond to different syntax forms such as abbreviations, use of acronyms or contextual information like country, sub-organization, etc. In order to address these potential variations the CORFU (Company, ORganization and Firm Unifier) approach seeks for providing a stepwise method to unify corporate names using NLP and semantics based techniques as a previous step to perform an entity reconciliation process. The execution of CORFU comprises several common but customized steps in natural language processing applications such as 1) text normalization; 2) filtering; 3) comparison and clusterization and 4) linking to an existing information resource. The CORFU unifier make an intensive use of the Python NLTK API and other packages for querying REST services or string comparison. Finally and due to the fact that the corporate name can change in each step the initial raw name must be saved as well as contextual information such as dates, acronyms or locations. Thus common contextual information can be added to create the final unified name.

1. Normalize raw text and remove duplicates. This step is comprised of: 1) remove strange characters and punctuation marks but keeping those that are part of a word avoiding potential changes in abbreviations or acronyms; 2) lowercase the raw text (although some semantics can be lost previous works and empirical tests show that this is the best approach); 3) remove duplicates and 4) lemmatize the corporate name. The implementation of this step to clean the corporate name has been performed using a combination of the aforementioned API and the Unix scripting tools AWK and SED. In this case, Figure 1 presents two relevant snippets of code for cleaning the name and making a basic word normalization.

```
rawname = filter(lambda x: x in string.letters or
                 x in string.whitespace, line)
...
def normalize(self, word):
  word = word.lower()
  word = self.lemmatizer.lemmatize(word)
  return word
```

Fig. 1. Normalization and data cleansing using the Python NLTK API.

2. Filter the basic set of common stopwords in English. A common practice in NLP relies in the construction of stopwords sets that can filter some non-relevant words. Nevertheless the use of this technique must consider two key-points: 1) there is a common set of stopwords for any language than can be often used as a filter and 2) depending on the context the set of stopwords should change to avoid filtering relevant words. In this particular case, a common and minimal set of stopwords in English provided by NLTK

has been used. Thus the normalized corporate name is transformed into a new set of words. Figure 2 presents the function for removing a set of words given a another set, it can also be applied to other stages that require filtering capabilities.

```
from nltk.corpus import stopwords
self.stop_words_wn = Set(stopwords.words('english'))
...
def remove_set(self, set,  name):
  token_names= word_tokenize(name)
  filtered_token_list = [w for w in  token_names if not w in set ]
  cleaned_name = "␣".join(["".join(filtered_token)
    for filtered_token in filtered_token_list])
  return cleaned_name

stop_unified_name = self.remove_set(self.stop_words_wn, name)
```

Fig. 2. Filtering words with the Python NLTK API.

3. Filter the expanded set of most common words in the dataset. Taking into account the aforementioned step this stage is based on the construction of a customized stopwords set for corporate names that is also expanded with Wordnet (ver. 3.0) synonyms with the aim of exploiting semantic relationships. In order to create this set two strategies have been followed and applied: 1) create the set of words by hand (accurate but very time-consuming) and 2) extract automatically the set of "most common words" from the working dataset and make a hand-validation (less accurate and time-consuming). Figure 3 partially shows these approaches implemented.
4. Dictionary-based expansion of common acronyms and filtering. A dictionary of common acronyms in corporate names such as "PTY", "LTD" or "PL" and their variants has been created by hand in order to be able to extract and filter acronyms.
5. Identification of contextual information and filtering. Mainly corporate names can contain nationalities or place names that, in most of cases, only add noise to the real corporate name. In this case, the use of external services such as Geonames, Google Places or Google Maps can ease the identification of these words and their filtering. In order to tackle this situation the REST web service of Geonames has been selected due to its capabilities to align text with locations.
6. Spell checking (optional). This stage seeks for providing a method for fixing misspelling errors. It is based on the well-known speller [24] of Peter Norvig that uses a train dataset for creating a classifier. Although the accuracy of this algorithm is pretty good for relevant words in corporate names, the

```
from nltk.corpus import wordnet as wn
...
def create_syns_from_wn(self, word):
  syns = wn.synsets(word)
  lemmas = Set()
  for syn in syns:
    lemmas = lemmas | Set([lemma.name for lemma in syn.lemmas] )
  return lemmas

def expand_list_wn(self,  list):
  source = Set(list)
  expanded_set = Set()
  for word in source:
    expanded_set = expanded_set | self.create_syns_from_wn(word)
return expanded_set |  source

def list_most_used_words(companies, max):
  words = flatten(map(lambda company: company.rawname.split(), companies))
  counter = collections.Counter(words)
  return [x[0].lower() for x in
  filter ( lambda x: x [1] > max,
   (itertools.islice(counter.most_common(), 0, 1000)))]
```

Fig. 3. Expanding a list of words with Wordnet syns and Counting "most used words" in a dataset.

empirical and unit tests with a working dataset have demonstrated that misspelled non-relevant words is more efficient and accurate using a stopwords set/dictionary (this set has been built with words that are not in the set of "most common words", step 2, and exist in the Wordnet database). Furthermore some spelling corrections are not completely adequate for corporate names due to words could change and, therefore, a non-acceptable variant of the name could be accidentally included. That is why this stage is marked as optional and must be configured and performed with extreme care.

7. Pos-tagging parts of speech according to a grammar and filtering the non-relevant ones. The objective of this stage lies in "classifying words into their parts of speech and labeling them accordingly is known as part-of-speech tagging" [17]. In order to perform this task both a lemmatizer based on Wordnet and a grammar for corporate names ("NN"-nouns and "JJ"-adjectives connected with articles and prepositions) have been designed, see Figure 4. Once words are tagged next step consists in filtering non-relevant categories in corporate names keeping nouns and adjectives, as an example Figure 4 also shows how to walk and filter nodes in the parsed tree.
8. Cluster corporate names. This task is in charge of grouping names by similarity applying a string comparison function. Thus if the clustering is applied n

```
self.lemmatizer = nltk.WordNetLemmatizer()
self.grammar = r"""
  NBAR: {<NN.*|JJ>*<NN.*>}
  NP: {<NBAR>}
      {<NBAR><IN><NBAR>}
    """
self.chunker = nltk.RegexpParser(self.grammar)

def leaves(self, tree):
  for subtree in tree.subtrees(filter = lambda t: t.node=='NP'):
    yield subtree.leaves()
```

Fig. 4. Regular expression-based chunker in Python NLTK and Filtering words by the category "NP" (noun phrase) .

times any name will be grouped by "the most probably/used name" according to a threshold generated by the comparison function. This first version of CORFU has used the WRatio function to compare strings (available in the Levenshtein Python package) and a custom clustering implementation.

9. Validate and reconcile the generated corporate name via an existing reconcile service (optional). This last step has been included with the objective of linking the final corporate name with an existing information resource and adding new alternative labels. The OpenCorporates and DBPedia reconciliation services have been used in order to retrieve an URI to new corporate name. As a consequence the CORFU unifier is partially supporting one of the main principles of the LOD initiative such as unique identification.

4 Use Case: Unifying supplier names in the Australian e-Procurement domain

As previous sections have introduced there is an increasing interest and commitment in public bodies to create a real transparent public administration. In this sense public administrations are continuously releasing relevant data in different domains such as tourism, health or public procurement with the aim of easing the implementation of new added-value services and improve their efficiency and transparency. In the particular case of public procurement, main and large administrations have already made publicly available the information with regards to public procurement processes. In this case of study the information of Australia is used to test the CORFU unifier. It is comprised of a dataset of more than 400K supplier names during the period 2004-2012. In order to be able to extract good statistics from this dataset the unification of names must be applied to. That is why the CORFU stepwise method has been customized to deal with the heterogeneities of this large dataset as Table 1 summarizes.

Step	Name	Customization
1	Normalize raw text and remove duplicates	Default
2	Filter the basic set of common stopwords in English	Default
3	Filter the expanded set of most common words in the dataset	Two stopwords sets: 355 words (manually) and words with more than $n = 50$ apparitions (automatically)
4	Dictionary-based expansion of common acronyms and filtering	Set of 50 acronyms variations (manually)
5	Identification of contextual information and filtering	Use of Geonames REST service
6	Spell checking (optional)	Train dataset of 128457 words provided by Peter Norvig's spell-checker [24].
7	Pos-tagging parts of speech according to a grammar and filtering the non-relevant ones	Default
8	Cluster corporate names	Default
9	Validate and reconcile the generated corporate name via an existing reconcile service (optional)	Python client and Google Refine

Table 1. Customization of the CORFU technique for Australian supplier names.

5 Evaluation

5.1 Research design

Since the CORFU approach has been designed and implemented [11] it is necessary to establish a method to assess quantitatively the quality of the results. The steps to carry out this experiment are: 1) Configure the CORFU technique, see Table 1; 2) Execute the algorithm taking as a parameter the file containing the whole dataset of company names; 3) Validate (manually) the dump of unified names; 4) Calculate the values of precision and recall. In particular, this evaluation considers the precision of the algorithm as "the number of supplier names that have been correctly unified under the same name" while recall is "the number of supplier names that have not been correctly classified under a proper name".

5.2 Sample

In order to validate the CORFU approach the dataset of supplier names in Australia in the period 2004-2012 containing 430188 full names and 77526 unique

[11] `https://github.com/chemaar/corfu`

names has been selected. The experiment has been carried out executing the aforementioned steps in the whole dataset to finally generate a dump containing for each supplier the raw name and the unified name. These mappings has been validated by hand to quantify the typical measures of precision and recall.

5.3 Results and Discussion

Total number of companies	**Unique names**	**CORFU unified names**	**% of unified names**	**Precision**	**Recall**	**F1 score**
430188	77526	40277	48%	0.762	0.311	0.441
430188	299 in 77526	68	100%	0.926	0.926	0.926

Table 2. Results of applying the CORFU approach to the Australian supplier names.

According to the results presented in Table 2, the precision and recall of the CORFU technique are consider acceptable for the whole dataset due to $77526 - 40278 = 37248$, a 48% of the supplier names, has been unified with a precision of 0.762 and recall of 0.311 (best values must be close to 1). The precision is good enough but the recall presents a low value because a good few of corporate names were not unified under a proper name due to some relevant words have been filtered.

In order to improve the results for relevant companies, the experiment has also been performed and evaluated for the first 100 companies in the Forbes list, actually 68 companies were found in the dataset. In this case, results show a better performance in terms of precision, 0.926, and recall,0.926, and all these supplier names, 299 in the whole dataset, were unified by a common right name. The explanation of this result can be found due to some of the parameters of the CORFU technique were specially selected for unifying these names because of their relevance in world economic activities.

On the other hand, it is important to emphasize that the last step of linking these names with existing web information resources using the reconciliation service of OpenCorporates or DBPedia in Google Refine has the potential of generating 28383 right links (36.61%) instead of the initial 8%. Thus the initial problem of linking (n string literals $\rightarrow$ 1 company $\rightarrow$ 1 URI) has been substantially improved.

Finally, the frequency distribution of supplier and number of appearances is depicted on Figures 5 and 6 with the objective of presenting how the cloud of points (appearances) that initially were only one per supplier has emerged due to the unification of names, for instance in the case of "Oracle" 75 apparitions can now be shown. On the other hand and due to the unique identification of supplier names, new RDF instances are generated, see Figure 7, and can be

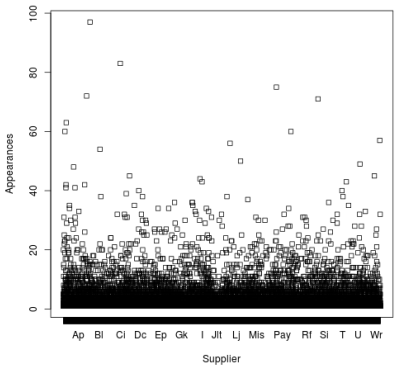

Fig. 5. Full view of supplier and number of appearances in the sample dataset.

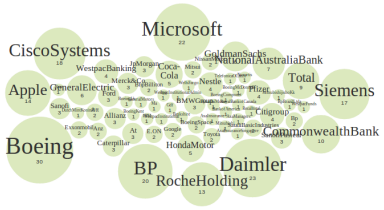

Fig. 6. Bubble Cloud of the first 100 Forbes companies and number of appearances in the sample dataset.

querying via SPARQL to make summary reports of the number of rewarding contracts by company, see Figure 8.

```
:o1 a org:Organization;
  skos:prefLabel
    ''Microsoft'';
  skos:altLabel
    ''Microsoft Australia'',
    ''Microsoft Australia
      Pty Ltd'',
    ...;
  skos:closeMatch
    dbpedia-res:Microsoft;
   ...
.
```

Fig. 7. Example of a RDF organization instance.

```
SELECT str(?label)
  (COUNT(?org) as ?pCount)
WHERE{
   ?ppn :rewarded-to ?org .
   ?org rdf:type org:Organization.
   ?org skos:prefLabel ?label.
   ...
}
GROUP BY str(?label)
ORDER BY desc(?pCount)
```

Fig. 8. Example of a SPARQL query for counting supplier names.

6 Conclusions and Future Work

A technique for unifying corporate names in the e-Procurement sector has been presented as a step towards the unique identification of organizations with the aim of accomplishing one of the most important LOD principles and easing the execution of reconciliation processes. The main conclusion of this work lies in

the design of a stepwise method to prepare raw corporate names in a specific context, e.g. Australia supplier names, before performing a reconciliation process. Although the percentage of potential right links to existing datasets has been dramatically improved it is clear that human-validation is also required to ensure the correct unification of names. As a consequence the main application of CORFU can be found when reporting or tracking activity of organizations are required. However this first effort has implied, on the one hand, the validation of the stepwise method and, on the other hand, the creation of a sample dataset that can serve as input for more advanced algorithms based on machine learning techniques such as classifiers. From public administrations point of view this technique also enables a greater transparency providing a simple way to unify corporate names and boosting the comparison of rewarded contracts.

Finally, further steps in this work consist in the extension of the stopwords sets for corporate names, a better acronym detection and expansion algorithm, other techniques to make string comparisons such as $n - grams$ and the creation of a new final step to enhance the current implementation with a classifier that can automatically learn new classes of corporate names. Furthermore the technique must be reported to the international "Public Spending" initiative, as supporting tool, to be applied over other datasets to correlate and exploit metadata of public contracts.

7 Acknowledgements

This work is part of the "PublicSpending.net" effort carried out in cooperation with Marios Meimaris (FIXME:affiliations), Giannis Xidias, Giorgos Vafeiadis and Michalis Klonaras.

References

1. Samur Araujo, Jan Hidders, Daniel Schwabe, and Arjen P De Vries. SERIMI – Resource Description Similarity , RDF Instance Matching and Interlinking. *WebDB 2012*, 2011.
2. Tim Berners-Lee. Linked data, jul 2006.
3. Breck Baldwin Bob Carpenter, Mitzi Morris. *Text Processing with Java 6*, volume 1. LingPipe Publishing, 2012.
4. Kalina Bontcheva, Hamish Cunningham, Ian Roberts, Angus Roberts, Valentin Tablan, Niraj Aswani, and Genevieve Gorrell. GATE Teamware: a web-based, collaborative text annotation framework. *Language Resources and Evaluation*, pages 1–23, 2013.
5. Directorate-General for Informatics European Commission. The eProcurement Map. a map of activities having an impact on the development of european interoperable eprocurement solutions, August 2011. `http://www.epractice.eu/en/library/5319079`.
6. Carmen Galvez and Félix Moya-Anegón. The unification of institutional addresses applying parametrized finite-state graphs (P-FSG). *Scientometrics*, 69(2):323–345, 2006.

7. Carmen Galvez and Félix Moya-Anegón. A Dictionary-Based Approach to Normalizing Gene Names in One Domain of Knowledge from the Biomedical Literature. *Journal of Documentation*, 68(1):5–30, 2012.
8. Kevin Gimpel, Nathan Schneider, Brendan O'Connor, Dipanjan Das, Daniel Mills, Jacob Eisenstein, Michael Heilman, Dani Yogatama, Jeffrey Flanigan, and Noah A. Smith. Part-of-speech tagging for Twitter: annotation, features, and experiments. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers - Volume 2*, HLT '11, pages 42–47, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.
9. T. Heath and Christian Ch. Bizer. *Linked Data: Evolving the Web into a Global Data Space*, volume 1. Morgan & Claypool, 2011.
10. Robert Isele, Anja Jentzsch, and Christian Bizer. Silk Server - Adding missing Links while consuming Linked Data. In *COLD*, 2010.
11. Robert Isele, Anja Jentzsch, and Christian Bizer. Active Learning of Expressive Linkage Rules for the Web of Data. In *ICWE*, pages 411–418, 2012.
12. Dan Klein, Joseph Smarr, Huy Nguyen, and Christopher D. Manning. Named entity recognition with character-level models. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003 - Volume 4*, CONLL '03, pages 180–183, Stroudsburg, PA, USA, 2003. Association for Computational Linguistics.
13. Michael Krauthammer and Goran Nenadic. Term identification in the biomedical literature. *J. of Biomedical Informatics*, 37(6):512–526, December 2004.
14. Stanford Natural Language Processing Lecture. Apache OpenNLP Developer Documentation, March 2013. `http://opennlp.apache.org/documentation/manual/opennlp.html`.
15. Stanford Natural Language Processing Lecture. Spelling Correction and the Noisy Channel. The Spelling Correction Task, March 2013. `http://www.stanford.edu/class/cs124/lec/spelling.pdf`.
16. Chenliang Li, Jianshu Weng, Qi He, Yuxia Yao, Anwitaman Datta, Aixin Sun, and Bu-Sung Lee. TwiNER: named entity recognition in targeted twitter stream. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '12, pages 721–730, New York, NY, USA, 2012. ACM.
17. Edward Loper and Steven Bird. NLTK: The Natural Language Toolkit. In *Proceedings of the ACL Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics*, pages 62–69. Somerset, NJ: Association for Computational Linguistics, 2002. `http://arXiv.org/abs/cs/0205028`.
18. Francisco J. López-Pellicer, Mário J. Silva, Marcirio Silveira Chaves, F. Javier Zarazaga-Soria, and Pedro R. Muro-Medrano. Geo Linked Data. In *DEXA (1)*, pages 495–502, 2010.
19. Fadi Maali, Richard Cyganiak, and Vassilios Peristeras. Re-using Cool URIs: Entity Reconciliation Against LOD Hubs. In Christian Bizer, Tom Heath, Tim Berners-Lee, and Michael Hausenblas, editors, *LDOW*, CEUR Workshop Proceedings. CEUR-WS.org.
20. Pablo N. Mendes, Max Jakob, Andrés García-Silva, and Christian Bizer. DBpedia spotlight: shedding light on the web of documents. In *Proceedings of the 7th International Conference on Semantic Systems*, I-Semantics '11, pages 1–8, New York, NY, USA, 2011. ACM.

21. Fernanda Morillo, Javier Aparicio, Borja González-Albo, and Luz Moreno. Towards the automation of address identification. *Scientometrics*, 94(1):207–224, January 2013.
22. David Nadeau. *Semi-Supervised Named Entity Recognition: Learning to Recognize 100 Entity Types with Little Supervision*. PhD thesis, School of Information Technology and Engineering, University of Ottawa, Ottawa, Canada, 2007.
23. David Nadeau and Satoshi Sekine. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30(1):3–26, January 2007.
24. Peter Norvig. How to Write a Spelling Corrector, March 2013. `http://norvig.com/spell-correct.html`.
25. K. Rafał. *Apache Solr 3.1 Cookbook*. Packt Publishing, Limited, 2011.
26. L. Ratinov and E. Gudes. Abbreviation Expansion in Schema Matching and Web Integration. In *Proceedings of the 2004 IEEE/WIC/ACM International Conference on Web Intelligence*, WI '04, pages 485–489, Washington, DC, USA, 2004. IEEE Computer Society.
27. Jesse Read, Albert Bifet, Geoff Holmes, and Bernhard Pfahringer. Scalable and efficient multi-label classification for evolving data streams. *Machine Learning*, 88(1-2):243–272, 2012.
28. Jose María Álvarez Rodríguez, José Emilio Labra Gayo, Francisco Adolfo Cifuentes Silva, Giner Alor-Hernández, Cuauhtémoc Sánchez, and Jaime Alberto Guzmán Luna. Towards a Pan-European E-Procurement Platform to Aggregate, Publish and Search Public Procurement Notices Powered by Linked Open Data: the Moldeas Approach. *International Journal of Software Engineering and Knowledge Engineering*, 22(3):365–384, 2012.
29. Luís Sarmento, Alexander Kehlenbeck, Eugénio Oliveira, and Lyle Ungar. An Approach to Web-Scale Named-Entity Disambiguation. In *Proceedings of the 6th International Conference on Machine Learning and Data Mining in Pattern Recognition*, MLDM '09, pages 689–703, Berlin, Heidelberg, 2009. Springer-Verlag.
30. Yefeng Wang. Annotating and recognising named entities in clinical notes. In *Proceedings of the ACL-IJCNLP 2009 Student Research Workshop*, ACLstudent '09, pages 18–26, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics.
31. Stuart Yeates. Automatic Extraction of Acronyms from Text. In *University of Waikato*, pages 117–124, 1999.

Appendix I

Raw Supplier Name	Target (potential) Supplier Name and URI
"Accenture" "Accenture Aust Holdings" "Accenture Aust Holdings" "Accenture Aust Holdings Pty Ltd" "Accenture Australia Holding P/L" "Accenture Australia Limited" ... "Accenture Australia Ltd"	"Accenture" http://live.dbpedia.org/resource/Accenture
"Microsoft Australia" "Microsoft Australia Pty Ltd" ... "Microsoft Enterprise Services"	"Microsoft" http://live.dbpedia.org/resource/Microsoft
"Oracle (Corp) Aust Pty Ltd" "Oracle Corp (Aust) Pty Ltd" "Oracle Corp Aust Pty Ltd" "Oracle Corp. Australia Pty.Ltd." "Oracle Corporate Aust Pty Ltd" "Oracle Corporation" "Oracle Risk Consultants" "ORACLE SYSTEMS (AUSTRALIA) PTY LTD" ... "Oracle University"	"Oracle" http://live.dbpedia.org/resource/Oracle_Corporation
"PRICEWATERHOUSECOOPERS(PWC)" "PricewaterhouseCoopers Securities Ltd" "PricewaterhouseCoopers Services LLP" "Pricewaterhousecoopers Services Pty Ltd" "PriceWaterhouseCoopers (T/A: PriceWaterhouseCoopers Legal)" ... "Pricewaterhouse (PWC)"	"PricewaterhouseCoopers" http://dbpedia.org/resource/PricewaterhouseCoopers
...	...

Table 3. Examples of supplier names in the Australian rewarded contracts dataset.