Building a Corpus-Derived Gazetteer for Named Entity Recognition

Norshuhani Zamin and Alan Oxley

Department of Computer and Information Sciences
Universiti Teknologi PETRONAS,
31750 Tronoh, Perak, Malaysia

Abstract. Gazetteers, or entity dictionaries, are an important element for Named Entity Recognition. Named Entity Recognition is an essential component of Information Extraction. Gazetteers work as specialized dictionaries to support initial tagging. They provide quick entity identification thus creating richer document representation. However, the compilation of such gazetteers is sometimes mentioned as a stumbling block in Named Entity Recognition. Machine learning, both rule-based and look-up based approaches, are often used to perform this process. In this paper, a gazetteer developed from MUC-3 annotated data for the 'person named' entity type is presented. The process used has a small computational cost. We combine rule-based grammars and a simple filtering technique for automatically inducing the gazetteer. We conclude with experiments to compare the content of the gazetteer with the manually crafted one.

Keywords: Gazetteer, Named Entity Recognition, Natural Language Processing, Terrorism.

1 Introduction

Named Entity Recognition (NER) involves the identification of certain occurrences of words or expressions in unstructured texts and the classification of them into a set of predefined categories of interest. NER is often implemented as a pre-processing tool for an Information Extraction (IE) system. One application is document retrieval, or automatic document forwarding. For this application [1] states that "… documents annotated with NE information can be searched more accurately than raw text." As an example, NER annotation would allow the search for all texts mentioning the *company* "Hong Leong" such as the notable "Hong Leong Bank" and "Hong Leong Group", both Malaysian establishments. NER is supposed to ignore documents about unrelated companies of the same name. However, our investigations found that classification of certain entities often involves challenging ambiguities. Among the types of ambiguity is metonymy [2]. Metonymy is a figure of speech in which one word or phrase is substituted for another with which it is closely related; for instance, "England" is an *organization* in the statement "England won the world cup" while "England" is a *location* in the statement "The world cup took place in England". The use of syntactic features and word similarity is a possible solution for metonymy recognition, as described in [2].

J.M. Zain et al. (Eds.): ICSECS 2011, Part II, CCIS 180, pp. 73–80, 2011.

NER systems are commonly classified into three categories: machine learning, rule-based and look-up based [3]. The machine learning approach requires an annotated training corpus to establish the learning process thus allowing it to predict the most likely entities in the given text. The machine learning approach has proved to be advantageous when there are either no terminologies or only partial ones. All of the basic machine learning techniques - supervised, semi-supervised and un-supervised - are only possible with the availability of huge amounts of training data. Nonetheless, the training data or corpus, which is often manually annotated, can be a really cumbersome task for a human to create.

Rule-based NER systems achieve the best results of the three categories in all NER evaluations. In the absence of learning/training data, rule-based NER shows promising results. However, the rules involved in rule-based NER are likely to be quite complex. This approach relies heavily on the knowledge of linguistic experts. Consequently, the hand crafted rules are difficult to maintain without help from the experts in supporting large-scale NER [4]. The look-up based approach is a straightforward method and provides an extremely efficient way to perform NER [5]. In general, the look-up based approach makes use of lists of common entities to provide clues. (The term "list" is often used interchangeably with the term "gazetteer", "lexicon" and "dictionary" [4].) It is a fast method with a small programming cost. The only processing required is to map the entities in the list against the given text. Additionally, lists are commonly acquired either from a corpus, the web or Wikipedia. However, NER systems based solely on such lists suffer from the limitations of coverage and ambiguity [6]; for instance, the word "Washington" in "President Washington" is clearly a *person* based on the external evidence as the regular expression "President" appears before the word. However, if the word "Washington" is found in the look-up list as a *location* (as in "Washington D.C."), then this entity will be identified as a location based on this internal evidence. This limitation shows that huge lists may also miss some important entities, but with proper techniques and methods the drawback is fixable.

In this paper, we investigate look-up based NER. The list to be looked up is referred to as a "gazetteer" throughout the paper. We propose a technique to automatically derive the gazetteer for name entities from a terrorism text corpus (MUC-3). Our experiment explored only the *person* entity type, by identifying the proper name, and the result is compared with the manually annotated list. We used a simple tokenizing technique for the entity extraction, with a dictionary filtering scheme inspired by [7], and combined it with simple lexical patterns [8] for the grammar rules.

2 Related Work

The task of automatically generating gazetteers for NER has been studied for many years. [9] uses lexical patterns to identify nouns from a similar semantic class. For instance, a noun phrase that follows "the President of" is usually the name of a country. The construction of such noun phrases are based on common patterns developed manually. These patterns are also referred to as grammar rules. Similarly, research in [10] uses lexical patterns but with a small number of entities called *seeds* to train the proposed bootstrapping algorithm in order to derive more related entities. Research in [5] shows that the use of a simple filtering technique for improving the automatically acquired gazetteer has contributed to a highly significant result, close to that of a state-of-the-art NER. A novel method for exploiting repetition of entities is presented in [7] with a highly efficient filtering technique to filter unwanted entities. The recall-enhancing approach requires an entire test set to be available despite substantially improving the extraction performance. Inspired by the work in [11], the researcher in [4] proposes an automatic approach to generate gazetteers based on initially defined entities *(seeds).* The bootstrapping algorithm in [10] is applied, with little modification, to handle novel types of named entities including car brands.

The use of the structural information of a language has recently been studied as one of the permissible approaches to automatically induce gazetteers from texts. The research in [12] shows a successful attempt at generating gazetteers from a Japanese corpus using the cache features, coreference relations, syntactic features and case-frame features of the language. This information is commonly available from structural analysis done by linguists. It has been observed from the evaluation results that the use of language specific features improves the performance of the system. Meanwhile, a novel method using the significant high-frequency strings of the corpus is introduced in [13]. The method uses the distribution of these strings in the document as candidate entities to filter the invalid entities. Additionally, the research team extends the work by incorporating word-level features and has successfully induced a gazetteer from Chinese news articles at around 80% accuracy.

More recently, Natural Language Processing research in [14] uses concepts and instance information from ontologies for NER and IE systems. The research automatically generates gazetteers from a corpus using the `rdf:type` information. An RDF stylesheet is defined and used to select statements about instances of relevant concepts. These instances are then converted to structured gazetteer source files. To the best of our knowledge, none of the discussed research generates gazetteers from the MUC-3 text corpus.

3 Corpus

A text corpus is a collection of text. Most corpora are designed to contain a careful balance of material in one or more genres. Commonly, information in corpora is unstructured. There is a wide range of corpora available, such as the collections of speeches[1], e-books[2], newswire articles[3] and texts of multiple genres. The Brown Corpus[4], which was established in 1961, is the pioneer corpus. It is a collection of 500 English text sources categorized by different genres. Some of these corpora contain linguistic annotations, representing part-of-speech tags, named entities, syntactic

[1] http://www.tlab.it/en/allegati/esempi/ inaugural.htm

[2] http://www.gutenberg.org

[3] http://www.reuters.com

[4] http://mailman.uib.no/public/corpora/2005-June/001262.html

structures, semantic roles, etc. However, most of the annotated corpora are not publicly accessible, such as the British National Corpus[5].

In this research we work with a terrorism corpus to support our ongoing research, which aims to develop a counterterrorism IE mechanism [15-17]. The series of Message Understanding Conferences (MUCs) funded by the Defense Advanced Research Projects Agency (DARPA) has established seven types of corpora, two of which are collections of American terrorism texts. The goal of these conferences was to encourage the development of new and better methods of IE. We use the MUC-3 corpus, a collection of news records on terrorism from Latin America. Unfortunately, this corpus is unannotated. The pre-processing part of our gazetteer generation process relies on the part-of-speech tags of the words to identify a possible group for an entity. A free part-of-speech tagger is recycled to perform the tagging process.

4 Gazetteer Generation

The framework of our automatic corpus-based gazetteer generation process is illustrated in Fig. 1. In the first step, to avoid starting from scratch, we adopted a free part-of-speech (POS) tagger known as Brill's Tagger [18] to assign possible POS tags to the words in the MUC-3 text corpus. Next, each word and its corresponding POS tag is tokenized by the *Word/Tag Tokenizer* module. The *Entity Extractor* module extracts all the words that have been assigned a 'proper noun singular' or 'proper noun plural' tag, as represented by NNP and NNPS, respectively, in the Brill's Tagger notation. Additionally, grammar rules are applied to potentially disambiguate the problem discussed earlier. The grammar rules adopted in [8] are the common lexical patterns or the regular expressions found in English text, i.e. the context around the proper names that identifies their type. Following are several regular expressions used to identify a person's name:

1. @Honorific CapitalizedWord CapitalizedWord
   a. @Honorific is a list of honorific titles such as General, Lieutenant, Captain, etc.
   b. Example: General Ramon Niebels
2. CapitalizedWord CapitalLetter @PersonVerbs
   a. @PersonVerbs is a list of common verbs that are strongly associated with people such as *{*said, met, walked, etc.*}*
3. @FirstNames CapitalizedWord
   a. @FirstNames is a list of common first names collected from the corpus.
   b. Example: Maza Marquez
4. CapitalizedWord CapitalizedWord [,] @PersonSuffix
   a. @PersonSuffix is a list of common suffixes such as *{*Jr., Sr., II, III, etc.*}*
   b. Example: Mark Green, Jr.
5. CapitalizedWord CapitalLetter [.] CapitalizedWord
   a. CapitalLetter followed by an optional period is a middle initial of a person and a strong indicator that this is a person name.
   b. Example: President Peace R.Pardo

[5] http://www.natcorp.ox.ac.uk

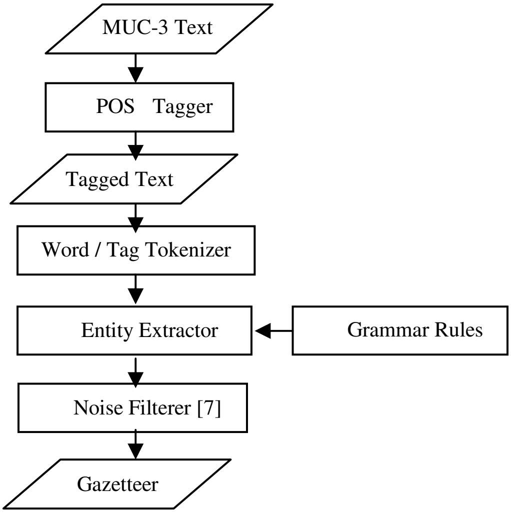

Fig. 1. Gazetteer Generation Framework

A dictionary matching scheme is often vulnerable to false positives. A false positive is a case where some proper names identified by the entity extractor are in fact non-names and can be considered as noise. False positives often degrade such a system's accuracy. Hence, we added the *Noise Filterer* module to the framework to remove the unwanted names by simply eliminating low-confidence predictions.

There are two metrics used in this module as introduced in [7]: Predicted Frequency (PF); Inverse Document Frequency (IDF). The PF metric estimates the degree to which a word appears to be used consistently as a name throughout the corpus.

$$PF(w) = \frac{cpf(w)}{ctf(w)} \tag{1}$$

Here, *cpf(w)* is the number of times that a word *w* is identified as a name and *ctf(w)* is the number of times it appears in the entire test corpus. Inverse Document Frequency is calculated using the IDF metric.

$$IDF(w) = \frac{\log(\frac{N+0.5}{df(w)})}{\log(N+1)} \tag{2}$$

Here, *df(w*) is the number of documents that contain the word *w* and *N* is the total number of documents in the corpus. *IDF* is a suitable metric for person name recognition since this type of entity does not frequently appear as an English word. Both metrics return a result between 0 and 1. A measure which combines these two metrics multiplicatively, giving a single probability of a word being a name and how common it is in the entire corpus, is as follows:

$$PF.IDF(w) = PF(w)xIDF(w) \tag{3}$$

A word with low *PF.IDF* score is considered ambiguous in the corpus and is excluded from the gazetteer.

5 Experimental Results

The system was evaluated using the *Precision (P)* and *Recall (R)* metrics. Briefly, *Precision* is the proportion of names proposed by the system which are true names while *Recall* is the proportion of true names which are actually identified. These metrics are often combined and referred to as the *F-Measure (F)*. Hence, the *F-Measure* is a weighted harmonic between *P* and *R*.

$$Precision\ (P) = correct\,/\,(correct + wrong) \tag{4}$$

$$Recall\ (R) = correct\,/\,(correct + missed) \tag{5}$$

$$F\text{-}Measure\ (F) = 2PR\,/\,(P + R) \tag{6}$$

Here, *correct* is the number of names extracted by the system that are persons names, *wrong* is the number of names extracted by the system that are not persons names while *missed* is the number of persons names that are extracted manually but not by the system. Our experiment was conducted on 40 randomly selected texts from the MUC-3 corpus. The same set of data is used on the Stanford NER [19]. The Stanford Named Entity recognizer uses Conditional Random Field (CRF) method and was trained on Conference on Computational Natural Language Learning (CoNLL), MUC-6, MUC-7 and Automatic Content Extraction (ACE) named entity corpora with a fairly results across domains. CRF is a type of discriminative undirected probabilistic graphical model which each vertex represents a random variable whose distribution is to be inferred. Edges correspond to dependencies between two random variables. The result of the performance evaluation for *person name* entity using MUC-3 text corpus is tabled.

Table 1. Performance Evaluation Results

System	Precision	Recall	F-Measure
Look-up based NER	0.79	0.85	0.81
Stanford NER	0.34	0.17	0.23

As can be seen, the results clearly indicates that our system outperform the Stanford parser. This experiment found that the Exact Match evaluation method used in CoNLL considers correct entities only if they are exactly match with the corresponding entities in the key tests [4]. Additionally, CoNLL dataset is a collection of newswire articles of general genres. Stanford NER was also trained on the MUC dataset as mentioned earlier: 1) MUC-6 is a collection of newswire articles on negotiation of labor disputes and corporate management succession and 2) MUC-7 is a collection of newswire articles on airplane crashes, rocket and missile launches. These are copyright dataset of the North American News Text Corpora which totally different genres and structures than the MUC-3. This shows that our look-up based NER performs better in MUC-3 dataset due to the use of the regular expression rules related to terrorism texts and the *Noise Filterer* module to filter unwanted *person name.*

However, our system achieved an average level of name recognition with 79% precision, 85% recall and an F-Measure of 81% as compared to human extraction. Technically, it is clear that having smaller test data and limiting lookups to noun phrases, as opposed to sentences, is undesirable. Our data observation found that it would be impossible to identify the name "Peruvian Landazuri" and "Pelito Landazuri" from the snippet "Peruvian and Pelito Juan Landazuri". Long names are often used by the Latin American people. In addition, due to the limited grammar rules, the system was likely to identify an organization as a person in cases such as "Adam Ltd" and "Phoenix Steel".

6 Conclusion

A domain-specific gazetteer often relies on domain-specific knowledge to improve system performance. It is generally agreed that domain-specific entities in technical corpora, such as the terrorism one, are much harder to recognize and the results have been less satisfactory than anticipated. This is due to the built-in complexity of terms in different domains which comprise of multiword expressions, spelling variations, acronyms, ambiguities, etc. The results indicate that our proposed approach is still immature and needs further improvement. The major drawbacks, in particular, are the limited test data, the non-specific grammar rules and the multiple occurrences of names in documents. In future, improvements will be made to address these weaknesses, to increase accuracy, as well as to identify entities that are more specifically related to counterterrorism, such as *location*, *weapon*, *tactic*, *transportation*, *type of document*, etc. These also include the cardinal entities such as *date*, *time*, *number of fatalities*, *number of injuries* and *money*.

References

1. Mikheev, A., Moens, M., Grover, C.: Name Entity Recognition without Gazetteers. In: 9th Conference of European Chapter of the Association of Computational Linguistic, pp. 1–8 (1999)
2. Nissim, M., Markert, K.: Syntactic Features and Word Similarity for Supervised Metonymy Resolution. In: 10th Conference of European Chapter of the Association of Computational Linguistic, pp. 56–63 (2003)
3. Tanenblatt, M., Coden, A., Sominsky, I.: The ConceptMapper Approach to Named Entity Recognition. In: 7th Language Resource and Evaluation Conference, pp. 546–551 (2010)

4. Nadeau, D.: Semi-Supervised Named Entity Recognition: Learning to Recognize 100 Entity Types with Little Supervision. PhD Thesis, University of Ottawa, Canada (2007)
5. Stevenson, M., Gaizauskas, R.: Using Corpus-derived Name Lists for Named Entity Recognition. In: North American Chapter of Association for Computational Linguistics, pp. 290–295 (2000)
6. Ratinov, L., Roth, D.: Design Challenges and Misconceptions in Named Entity Recognition. In: 31th Conference on Computational Natural Language Learning, pp. 147–155 (2009)
7. Minkov., E., Wang, R.C., Cohen, W.W.: Extracting Personal Names from Email: Applying Named Entity Recognition to Informal Text. In: Human Language Technology / Empirical Methods in Natural Language Processing, pp. 443–450 (2005)
8. Feldman, R., Sanger, J.: The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data. Cambridge University Press, Cambridge (2006)
9. Hearst, M.: Automatic Acquisition of Hyponyms from Large Text Corpora. In: International Conference on Computational Linguistics, pp. 539–545 (1992)
10. Riloff, E., Jones, R.: Learning Dictionaries for Information Extraction using Multi-level Bootstrapping. In: 16th National Conference on Artificial Intelligence, pp. 474–479 (1999)
11. Etzioni, O., Cafarella, M., Downey, D., Popescu, D., Shaked, A.M., Soderland, T., Weldnad, D.S., Yates, A.: Unsupervised Named Entity Extraction from the Web: An Experimental Study. J. Artificial Intelligence 165, 91–134 (2005)
12. Sasano, R., Kurohashi, S.: Japanese Named Entity Recognition using Structural Natural Language Processing. In: 3rd International Joint Conference on Natural Language Processing, pp. 607–612 (2008)
13. Pang, W., Fan, X., Gu, Y., Yu, J.: Chinese Unknown Words Extraction Based on Word-Level Characteristics. In: 9th International Conference on Hybrid Intelligent System, pp. 361–366 (2009)
14. Krieger, H.U., Schäfer, U.: DL Meet FL: A Bidirectional Mapping between Ontologies and Linguistic Knowledge. In: 23rd International Conference on Computational Linguistics, pp. 588–596 (2010)
15. Zamin, N., Oxley, A.: Information Extraction for Counter-Terrorism: A Survey on Link Analysis. In: International Symposium on Information Technology, pp. 1211–1215 (2010)
16. Zamin, N., Oxley, A.: Unapparent Information Revelation: A Knowledge Discovery using Concept Chain Graph Approach. In: National Seminar on Computer Science and Mathematics (2010) (Internal Publication)
17. Zamin, N.: Information Extraction for Counter-Terrorism: A Survey. Computation World: Future Computing, Service Computation, Cognitive, Adaptive, Content, Patterns, 520–526 (2009)
18. Brill, E.: Transformation-Based Error-Driven Learning and Natural Language Processing: A Case Study in Part of Speech Tagging. J. Computational Linguistics 21(4), 543–556 (1995)
19. Finkel, J.R., Grenager, T., Manning, C.: Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling. In: 43rd Annual Meeting of the Association for Computational Linguistics (ACL 2005), pp. 363–370 (2005)