Semantic Extraction of Named Entities from Bank Wire text

Ritesh Ratti
Pitney Bowes Software
Noida India
ritesh.ratti@pb.com

Himanshu Kapoor
Pitney Bowes Software
Noida India
himanshu.kapoor@pb.com

Shikhar Sharma
Pitney Bowes Software
Noida India
shikhar.sharma@pb.com

Anshul Solanki
Pitney Bowes Software
Noida India
anshul.solanki@pb.com

Pankaj Sachdeva
Pitney Bowes Software
Noida India
pankaj.sachdeva@pb.com

Abstract

Online transactions have increased dramatically over the years due to rapid growth in digital innovation. These transactions are anonymous therefore user provide some details for identification. These comments contain information about entities involved and transfer details which are used for log analysis later. Log analysis can be used for fraud analytics and detect money laundering activities. In this paper, we discuss the challenges of entity extraction from such kind of data. We briefly explain what wired text is, what are the challenges and why semantic information is required for entity extraction. We explore why traditional IE approaches are in-sufficient to solve the problem. We tested the approach with available open source tools for Entity extraction and describe how our approach is able to solve the problem of entity identification.

1 Introduction

Named Entity Extraction is the process of extracting entities like Person, Location, Address, Organization etc. from natural language text. However, named entities might also exist in non-natural text like Log data, Bank transfer content, Transactional data etc. Hence we require a system which should be robust enough to deal with the issues such as degraded and un-structured text rather than natural language text with correct spelling, punctuations and grammar. Existing information extraction methods are not able to deal with these requirements as most of the information extraction tasks work over natural language text. Since the context of language is missing in unstructured text, it is difficult to extract the entities from it and features are based on the natural language hence it requires semantic processing capabilities to understand the hidden meaning of content using dictionaries, ontologies etc.

Wire text is an example of such kind of text which is un-formatted and non-grammatic in nature. It can contain some letters in capital and some in small. For example people generally write the comments in short form and use multiple abbreviations. Bank wire text can be of this following format:

EVERITT 620122T NAT ABC INDIA LTD
REF ROBERT REASON SHOP RENTAL
REF 112233999 - REASON SPEEDING FINE
GEM SS HEUTIGEM SCHIENDLER
PENSION CH1234 CAB28

There are two major challenges in creating the machine learning model for wire text :

- Non-availability of data set due to confidentiality
- Non-contextual representation of text

In: Proceedings of IJCAI Workshop on Semantic Machine Learning (SML 2017), Aug 19-25 2017, Melbourne, Australia.

To identify the entities from such kind of text, it is therefore required special pre-processing of the text using semantic information of content. In this paper, we discuss the solution to extract entities from such kind of text. We evaluate our approach for Bank wire transfer text and make use of wordnet taxonomy for identifying the semantics for each of keyword. This paper is arranged in following sections. In Section 2 we discuss available methods of entity extraction. In Section 3 we describe the algorithm in detail and components involved. Section 4 we show the experimentation results and comparison with open source utilities. Section 5 is for conclusion & future work.

2 Background

Supervised machine learning techniques are primary solutions to solve the named entity recognition problem which requires data to be annotated. Supervised methods either learn disambiguation rules based on discriminative features or try to learn the parameter of assumed distribution that maximizes the likelihood of training data. Conditional Random fields [SM12] is the discriminative approach to solve the problems which uses sequence tagging. Other supervised learning models like Hidden Markov Model (HMM) [RJ86], Decision Trees, Maximum Entropy Models (ME), Support Vector Machines (SVM) also used to solve the classification problem. HMM is the earliest model applied for solving NER problem by Bikel [BSW99] for English. Bikel introduced a system, IdentiFinder, to detect NER using HMM as a generative model. Curran and Clark [CC03] applied the maximum entropy model to the named entity recognition problem. They used the softmax approach to formulate. McNamee and Mayfield [MMP03] tackle the problem as a binary decision problem, i.e. if the word belongs to one of the 8 classes, i.e. B- Beginning, I- Inside tag for person, organization, location and misc tags, Thus there are 8 classifiers trained for this purpose. Because of unavailability of wire text, it is difficult to create the tagged content hence supervised approaches are not able to solve the problem.

Various unsupervised schemes are also proposed to solve the entity recognition problem. People suggest the gazetteer based approach which help in identifying the keywords from the list. KNOWITALL is such a system which is domain independent and proposed by Etzioni [ECD$^+$05] that extracts information from the web in an unsupervised, open-ended manner. It uses 8 domain independent extraction patterns to generate candidate facts. Manning [GM14] have proposed a system that generates seed candidates through local, cross-language edit likelihood and then bootstraps to make broad predictions across two languages, optimizing combined contextual, word-shape and alignment models.

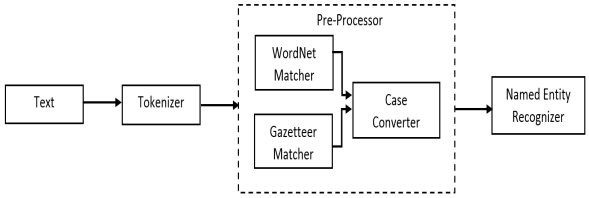

Figure 1: Component Diagram

Semantic Approaches also exists for named entity extraction. [MNPT02] used the wordnet specification to identify the WordClass and WordInstances list for each of the word to identify based on predefined rules. But that list is limited. [Sie15] uses word2Vec representation of words to define the semantics between words, that enhances the classification accuracy. It uses a continuous skipgram model which requires huge computation for learning word vectors. [ECD$^+$05] specifiy the gazetteer based feature as external knowledge for good performance. Given these findings, several approaches have been proposed to automatically extract comprehensive gazetteers from the web and from large collections of unlabeled text [ECD$^+$04] with limited impact on NER. Kazama [KT07] have successfully constructed high quality and high coverage gazetteers from Wikipedia.

In this paper, we propose the semantic disambiguation of named entities using wordnet and gazetteer. Our approach is based on pre-processing the text before passing it to Named entity recognizer.

3 Algorithm

3.1 Method

Named Entity Recognition involve multiple features related to the structural representation of entities hence proper case information imparts a valuable role in defining the entity type. For example : Person is generally written in Camel Case in english language & Organization are in Capitalized format. Our approach is based on orthogonal properties of entities. It is based on conversion of input data using wordnet after looking into the semantics for each of the word and providing existing NER the converted output. Now converted text is more probable to extract the Named entities once provided. We hereby propose the intermediate layer so called Pre-Processor as shown in Figure 1. Pre-Processor contains three major components called WordnetMatcher, GazetteerMatcher and

CaseConverter, whose purpose is to match the text efficiently with the given content list and converting the text to required case. LowerCaseConverter, CamelCaseConverter and UpperCaseConverter are instances of CaseConverter.
Tokenizer's main job is to convert the sentence into tokens. Named Entity Recognizer is used to extract the named entities.

We used Wordnet [Mil95] which provides the information about synsets. English version contains 129505 words organized into 99642 synsets . In wordnet two kinds of relations are distinguished: semantic relations (IS-A , part of etc.) which hold among synsets and lexical relations (synonymy , antonymy) which hold among words. Our gazetteer contains the dictionary for Person names, Organization names, Locations etc. Our approach work according to the following algorithm.

3.2 Approach

Algorithm 1: Semantic NER

Input : Sentence S as collection of words W and gazateers $List_{Names}$, $List_{Organization}$, $List_{Location}$, $List_{Ignore}$

Output: Set of entities $e_i \in E$

```
for each w_i ∈ S do
    w_i ← LowerCaseConverter(w_i)
    if w_i ∉ List_Ignore then
        synsets[] ← WordNetMatcher(w_i)
        if synsets[] ∉ Empty then
            if w_i ∈ List_Names then
                w_i ← CamelCaseConverter(w_i)
            end if
        else
            if w_i ∈ List_Organization or w_i ∈ List_Location
            then
                w_i ← UpperCaseConverter(w_i)
            else
                w_i ← CamelCaseConverter(w_i)
            end if
        end if
    end if
end for
(e_i) ← NamedEntityRecognizer(S)
```

Our algorithm works by looking up the pre-defined list in multiple steps. For each word in your input, first it converts to all lower-case, then check the word against the ignore list containing pronouns, prepositions, conjunctions and determiners. If it exists then we ignore the keywords. Else pass the lower-case-word to the WordNet API to get list of SynSets. If synsets are non-empty, such a word is likely to have some meaning so it will be checked with Names list first if found convert it to Camel Case like: John Miller , Robert Brown. If not found in namesList, later check in organization list and Location list. If match found convert to Upper Case otherwise convert in Camel Case. Now this pre-processed text is having meaningful representation of entities which is further passed to Named Entity Recognizer to extract the entities from the converted text.

3.3 Model Description

Our Named Entity Recognizer is based on Conditional Random Field [SM12], which is a discriminative model. We used cleartk library [BOB14] for model generation which uses mallet internally for implementation. Conditional random fields (CRFs) are a probabilistic framework for labeling and segmenting sequential data, based on the conditional approach.

Laferty [LMP$^+$01] define the the probability of a particular label sequence y given observation sequence x to be a normalized product of potential functions, each of the form .

$$\exp\ (\ \sum_j \lambda_j t_j(y_{i-1}, y_i, x, i) + \sum_k \lambda_k s_k(y_i, x, i)\ )$$

where $t_j(y_{i-1}, y_i, x, i)$ is a transition feature function of the entire observation sequence and the labels at positions i and $i-1$ in the label sequence; $s_k(y_i, x, i)$ is a state feature function of the label at position i and the observation sequence; and λ_j and μ_k are parameters to be estimated from training data.

When defining feature functions, we construct a set of real-valued features b(x, i) of the observation to expresses some characteristic of the empirical distribution of the training data that should also hold of the model distribution. An example of such a feature is : b(x, i) is 1 if observatuin at i is "Person" else 0

Each feature function takes on the value of one of these real-valued observation features b(x, i) if the current state (in the case of a state function) or previous and current states (in the case of a transition function) take on particular values. All feature functions are therefore real-valued. For example, consider the following transition function:

$$t_j(y_{i-1}, y_i, x, i) = \text{b(x,i)}$$

and ,

$$F_j(y, x) = \sum_{i=1}^{n} f_j(y_{i-1}, y_i, x, i)$$

Table 1: Features used for NER

Entity Type	Feature
Person	preceding = 1 succeeding = 2 , posTag , characterPattern , middleNamesList
Location	preceding = 3 succeeding = 3 , characterPattern , isCapital
Organization	preceding = 3 succeeding = 3 , posTag , characterPattern , orgSuffixList

where each $f_j(y_{i-1}, y_i, x, i)$ is either a state function $s_k(y_i, x, i)$ or a transition function $t(y_{i-1}, y_i, x, i)$. This allows the probability of a label sequence y given an observation sequence x to be written as

$$p(y|x, \lambda) = \frac{1}{Z(x)} \exp\ (\ \sum_j \lambda_j F_j(y, x)\ )$$

where Z(x) is a normalization factor.

3.4 Feature Extraction

We used multiple syntactic and linguistic features specific to entities. We also used pre-defined list match as a feature in couple of entities which improves the accuracy of our model. Our feature selection is based on following table 1. Explanation for the features is as follows :

- *Preceding*: Number of words to be considered for feature generation before the current word.
- *Succeeding*: Number of words to be considered for feature generation after the current word.
- *posTag* : Part of Speech tag as linguistic feature.
- *characterPattern* : Character pattern as feature in token like Camel Case, Numeric, AlphaNumeirc etc.
- *isCapital* : True if all the letters are in capitalized format.
- *xxxList* : Specific keyword list to match with the current word.True if word matches.For ex : orgSuffix contains list of suffixes used in organization names and middleNames consists the keywords used in middle name.

4 Experimentation Results

4.1 Dataset

We trained our NER model over MASC (Manually Annotated Sub-Corpus) dataset [PBFI12] which contains 93232 documents with 3232 different entities. We used the bank wire transfer text to verify the approach. Due to non-availability of bank wire text because of security reasons, We have to generate test set based on our client experience and understanding multiple user scenarios. We implemented the approach to our product [Pit] which is used by our clients.

Table 2: Comparison Results

Entity Type	Approach	Precision	Recall	Acc.
Person	Our Approach	0.65	0.306	0.27
	Stanford-NER	0.23	0.175	0.12
Location	Our Approach	0.88	0.57	0.53
	Stanford-NER	0.71	0.58	0.51
Organization	Our Approach	0.18	0.32	0.28
	Stanford-NER	0.03	0.018	0.012

4.2 Comparison

Our test dataset contains different types of comments which are non-natural in nature. We compare the approach with existing open source solutions like Open NLP [Apa14] and Stanford NER [MSB+14] and we justify that our approach works better due to the semantic conversion of the text. We observed that Open nlp is not able to detect much entities however Stanford NER is able to detect some of them. Table 2 describes the results of precision, recall and accuracy for entities Person, Location & Organization.

5 Conclusion & Future Work

We hereby proposed the approach for semantic conversion of bank wire text and extract the entities from converted text. Currently, we tested our approach for person, organization and location but it is easily extensible for other entities like address, contact number, email information etc. The approach uses semantic information from wordnet for preprocessing which can further be used to extract the entities from similar types of dataset like weblogs, DBlogs, transaction logs etc.

References

[Apa14] Apache Software Foundation. openNLP Natural Language Processing Library, 2014. http://opennlp.apache.org/.

[BOB14] Steven Bethard, Philip Ogren, and Lee Becker. Cleartk 2.0: Design patterns for machine learning in uima. In *Proceedings of the Ninth International Conference on Language Resources and Evalua-*

tion (LREC'14), pages 3289–3293, Reykjavik, Iceland, 5 2014. European Language Resources Association (ELRA). (Acceptance rate 61%).

[BSW99] Daniel M Bikel, Richard Schwartz, and Ralph M Weischedel. An algorithm that learns what's in a name. *Machine learning*, 34(1-3):211–231, 1999.

[CC03] James R. Curran and Stephen Clark. Language independent ner using a maximum entropy tagger. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003 - Volume 4*, CONLL '03, pages 164–167, Stroudsburg, PA, USA, 2003. Association for Computational Linguistics.

[ECD+04] Oren Etzioni, Michael Cafarella, Doug Downey, Ana-Maria Popescu, Tal Shaked, Stephen Soderland, Daniel S Weld, and Alexander Yates. Methods for domain-independent information extraction from the web: An experimental comparison. In *AAAI*, pages 391–398, 2004.

[ECD+05] Oren Etzioni, Michael Cafarella, Doug Downey, Ana-Maria Popescu, Tal Shaked, Stephen Soderland, Daniel S. Weld, and Alexander Yates. Unsupervised named-entity extraction from the web: An experimental study. *Artificial Intelligence*, 165(1):91 – 134, 2005.

[GM14] Sonal Gupta and Christopher D Manning. Improved pattern learning for bootstrapped entity extraction. In *CoNLL*, pages 98–108, 2014.

[KT07] Junichi Kazama and Kentaro Torisawa. Exploiting wikipedia as external knowledge for named entity recognition. 2007.

[LMP+01] John Lafferty, Andrew McCallum, Fernando Pereira, et al. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the eighteenth international conference on machine learning, ICML*, volume 1, pages 282–289, 2001.

[Mil95] George A. Miller. Wordnet: A lexical database for english. *Commun. ACM*, 38(11):39–41, November 1995.

[MMP03] James Mayfield, Paul McNamee, and Christine Piatko. Named entity recognition using hundreds of thousands of features. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003 - Volume 4*, CONLL '03, pages 184–187, Stroudsburg, PA, USA, 2003. Association for Computational Linguistics.

[MNPT02] Bernardo Magnini, Matteo Negri, Roberto Prevete, and Hristo Tanev. A wordnet-based approach to named entities recognition. In *Proceedings of the 2002 workshop on Building and using semantic networks-Volume 11*, pages 1–7. Association for Computational Linguistics, 2002.

[MSB+14] Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60, 2014.

[PBFI12] Rebecca J Passonneau, Collin Baker, Christiane Fellbaum, and Nancy Ide. The masc word sense sentence corpus. In *Proceedings of LREC*, 2012.

[Pit] Pitney Bowes Software CIM Suite http://www.pitneybowes.com/us/customer-information-management.html.

[RJ86] L. Rabiner and B. Juang. An introduction to hidden markov models. *IEEE ASSP Magazine*, 3(2):4–16, Jan 1986.

[Sie15] Scharolta Katharina Sienčnik. Adapting word2vec to named entity recognition. In *Proceedings of the 20th Nordic Conference of Computational Linguistics, NODALIDA 2015, May 11-13, 2015, Vilnius, Lithuania*, number 109, pages 239–243. Linköping University Electronic Press, 2015.

[SM12] Charles Sutton and Andrew McCallum. An introduction to conditional random fields. *Foundations and Trends in Machine Learning*, 4(1):267–373, 2012.