American Journal of Engineering Research (AJER)
e-ISSN: 2320-0847 p-ISSN : 2320-0936
Volume-9, Issue-5, pp-170-174
www.ajer.org
Research Paper Open Access

Named Entity Recognition in Vietnamese documents based on CRF

Vo Trung Hung
University of Technology and Education - The University of Danang, Vietnam
Corresponding Author: Vo Trung Hung

ABSTRACT: *Named Entity Recognition is a subfield of Information Extration andis getting wide attention. Researches with big languages such as English, French or Chinese produce good results but there are many limitations on languages that are not used very much, especially Vietnamese. The purpose of this study is building a named entity recognition system allowing identification of named entities such as person name, location, organization, time in Vietnamese texts by using CRF++ tool. The main task is creating tools and training data for building a named entity recognition model to facilitate the identification of the entities in vienamese documents. The Entity Recognition system was evaluated 10 times on over 300 documents and gives the average f-measure of 84,8%.*

Date of Submission: 08-05-2020 Date of acceptance: 22-05-2020

I. INTRODUCTION

In any text document, there are particular terms that represent specific entities that are more informative and have a unique context. These entities are known as named entities, which more specifically refer to terms that represent real-world objects like people, places, organizations, and so on, which are often denoted by proper names. A naive approach could be to find these by looking at the noun phrases in text documents. Named entity recognition (NER), also known as entity chunking/extraction, is a popular technique used in information extraction to identify and segment the named entities and classify or categorize them under various predefined classes [1].

NER is applied in many fields of natural language processing such as automatic question and answer system, machine translation, information query... At present, the identification of English is highly accurate due to the large data source, clear syntax [2], but for Vietnamese is still a challenge.

This article presents an overview of named entity recognition in Vietnamese text and uses the CRF (Condition Random Field) model, specifically CRF++ version 0.58, to identify entities.

The content of paper is organized into 4 sections.After the introduction, we present some research results related to this topic including entity identification and approaches to the problem of entity identification. Next, we present the building of a system of identification entity names for Vietnamese.At the end of the paper, we present some of the achieved results and development directions for the time being.

II. RELATED WORKS

A. Entity Identification

1) Information extraction

Information extraction is the name for techniques that extract structured information from unstructured text and render predefined information about entities and their relationships from text [3]. A number of levels of extracting information from a text include Entity Extraction, Relation Extraction, Co-reference Resolution. The scope of extracting is not only in words of the text, but also in sounds, images, ... The techniques used in extracting information include: segmentation, classification, combination and clustering.

2) Entity identification problem

Typically, each document contains objects such as names of people, organizations, places, dates, numbers, etc. These objects are collectively referred to as identification entities. The purpose of the entity identification problem is to identify these types of entities to assist us in understanding text. This is the most basic problem to consider before solving more complex problems in information extraction.

B. Approaches to the problem of entity identification

1) Knowledge-based approach

The knowledge-based approach (also called manual) is characterized by a hand-built law system that completely depends on the expert's own experience in each field. The rules always arise and it is constantly updated and put into the data warehouse under the strict censorship and correction of experts to have a complete entity identification system. A prime example is New York University's Proteous entity type recognition system participating in the MUC-6 [4] workshop supported by a large number of laws.

To build a system like the model above requires experts to have linguistic experience and a relatively large amount of time to implement the constantly updating new laws.

2) Approach based on machine learning

With the limitations of the approach to knowledge, the question posed to build a system that can "self-study" to make the system more flexible. There are several widely used and effective machine learning methods such as HMM, MEMM and CRF models.

HMM model (Hidden Markov Model) [5] was introduced and studied in the late 1960s and early 1970s. This is a finite state machine model with parameters representing the state transitions and probability of birth. Observed data at each state, is a statistical model in which the modeling system is thought to be a Markov process with unknown parameters and the task is to identify the hidden parameters from the parameter parameters. is based on this acknowledgment. The process of generating the observed data series in HMM through a series of state transitions, comes from one of the starting and stopping states in an ending state. The parameters of the model drawn can then be used to perform subsequent analyzes. With the problem of entity recognition, it is possible to view each of the corresponding states in one of the labels B-LOC, I-LOC, B-TIME, B-PER, etc. and observation data are words in the sentence. The sequence of the best described states for the observed data series can then be found by calculation.

$$P(S|O)=P(S,O)/P(O) \qquad (1)$$

In (1), S is the hidden state string, O is the known observed data series. Finding the sequence S* with probability P(S|O) reaches the maximum value is equivalent to finding S* making the maximum P(S,O).

The limitation of the Markov model lies in the fact that to calculate the probability of P(S,O), we usually have to list all possible cases of the S and O series. In fact, the Y series is finite and can be listed. statistics (O data) are very abundant. Besides, with some problems, the use of probability conditions P(S|O) gives better results.

The MEMM (Maximum Entropy Markov Models) model [6] assumes that observations are given and that we do not need to care about the probability of generating them, what is important here is the probability of transitions. For this model, the current observation does not exist independently, but rather relates to the state transitions, meaning that it depends on the previous state.

Probability P(S|O) can be calculated as follows:

$$P(S|O)=P(S_1,O_1)*\prod_{t=1}^{n}P(S_t|S_{t-1},O_t) \qquad (2)$$

MEMM considers observed data as a given condition instead of treating them as components generated by a model such as HMM, so the probability of transitions may depend on the diverse properties of the observed data series. These properties play an important role in determining the next state.

The CRF (Conditional Random Fields) model [7] was first introduced in 2001. This is a probability model for sequential labeling and data segmentation.

CRF is treated as a conditional scalar graph, where X is a random variable that receives values as the data series to be assigned, Y is a random variable that receives values as corresponding labels. In the entity identification problem, X can accept values as words in the text, Y is a random string of entity name labels (<LOCATION>, <ORGANIZE>, ...).

Let G = (V, E) be a non-periodic scalar graph and have vertices v ∈V corresponding to each random variable representing $Y_v$of Y. If each random variable Y_v follows the Markov property with the graph G then (Y, X) is a random field of CRF conditions.

$$P(Y_v|X,Y_w,\omega\neq v)=P(Y_v|X,Y_\omega,\omega\in N(v)) \quad (3)$$

where N (v) is a set of neighboring vertices of v.

In the simple but also important case, when modeling sequences, the graph G is represented as:

$$G=(V=\{1,2,...n\},\ E=\{(i,i+1)\})(4)$$

and can be illustrated in the following figure:

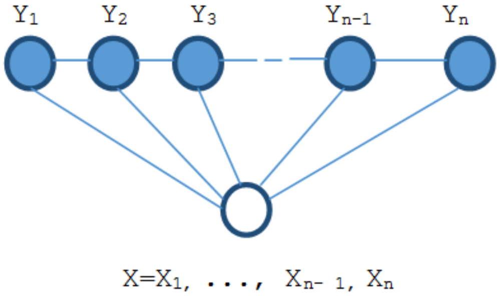


Fig.1. Scalar graph depicting CRF

Applying Markov's random field theory [8], the distribution of the Y-labeled series with a given X-observation series has the following form [7]:

$$p(y|x) \propto \exp(\sum_{e\in E,k} \lambda_k f_k(e, y|_e, x) + \sum_{e\in E,k} \mu_k g_k(v, y|_v, x))$$

where x is the observation string, y is the state series, y|S is the set of elements of y corresponding to the vertices of the subgraph S; f_k and g_k are self-defined attribute functions, λ_k and μ_k are parameters.

3) CRF++ Toolkit

This toolkit is developed based on CRF model. CRF++ is an open source tool written in C++ language and can serve for sequencing, labeling data sequentially ... This tool is running on the Windows operating system, it is included training and testing components.

During the training phase, a training file of CRF++ format is created and used. For each word in the text string, tags are identified, containing the word itself, a number of attributes and labels are assigned. Each card will be on a line of the training file. The properties at position i^{th} in the observation string consist of two parts: contextual information at position i^{th} and label information. Attribute selection is the selection of context samples that represent information of interest at any location in the observed data series. You can use context samples about word characteristics such as upper and lower case, numbers and punctuation marks; using the context form as a regular expression (example applies to defining time expressions); Using dictionary context allows lookup of words in a given list.

In addition to the training file, a template file (template) is used, which defines how to observe during training and testing. Each line in this sample file indicates a pattern used to define input data.

III. BUILDING A SYSTEM OF IDENTIFYING ENTITY NAMES IN VIETNAMESE DOCUMENTS

In this research, the system of identifying proper names in Vietnamese documents is built, including 2 components: training system and application of entity recognition. Software modules are written in the Java language.

A. Training

In the training system, we use the CRF++ toolkit and create training data including the following steps:

- Firstly, we build a dictionary database. It is included text files containing dictionaries of people, places, words before people's names, organizations, and time.

- Secondly, we create a training dataset.We collect and store manually text files. We use the vnTagger 4.2 tool to assign a category word label to the text and the result is a file containing keywords.

- Thirdly, we determine word attribute in the text which is done by software modules. On each line, the first column is the word itself, the next column is a category word label. Next, we create columns of attributes Is_Cap (uppercase), Is_Num (number), Is_Mark (punctuation), Is_Num (number), Is_4_Digit (4 digits), Is_Date (date value), Is_Family (family name), Is_Location (location), Is_BeforePER (from previous people's name), Is_BeforeORG (from previous organization name), Is_BeforeTime (from before time) in the next columns.

- Finally, we perform manually labeling in the last column whichare defined in the system as in Table 1:

Table 1. Types of labels

Label	Description
LOC	Name of the place
PER	Person name
ORG	Organization Name
NUM	Number

CUR	Currency
TIME	Time
PCT	Percentage
MISC	Other entities
O	Not entity

For multi-syllable words, the prefixes are used to determine the position of the sound (syllable) in the word (B: start, I: inside and O: end word). For example, the word "Thua Thien Hue" will correspond to the following 3 tags:Thừa <B-LOC>; Thiên<I-LOC>; Huế<E-LOC>.

Because CRF ++ does not support Vietnamese encoding, the resulting text file contains the tag and label attribute columns converted into Telex encoded Vietnamese (e.g. Vietnamese encoded into Vieejt) . As a result, the train.data file was created for use with the crf_learn.exe training tool to create model.data model file.

B. Training data extension

After the first entity identification model was created, the test data was collected, automatically identified and converted to a Telex code format similar to the test data to create the test.data file, however, manual labeling is not performed. Instead, we use the crf_test.exe testing tool of CRF++ to automatically label the last column. Next, this file is checked manually and corrected for errors to ensure accuracy. The test.data file data is then added to train.data to repeat the training process. The process of testing - supplementing on this training data is repeated a number of times to increase the reliability of the model.

C. Testing

To evaluate the performance of the entity identification system three parameters of precision, recall, and f-measure are used.

- Precision is measured as a percentage of the number of correctly labeled entities (value t_1) of the total number of labeled entities (t_2 value):

$$\text{Precision}=\frac{t_1}{t_2}$$

- Recall is measured as the percentage of correctly labeled entities (value t_1) of the total number of labeled entities of the CRF++ tool in test.data (t_3 value):

$$\text{Recall}=\frac{t_1}{t_3}$$

- F-measure is the quantity calculated by the combination of accuracy and recall by the following formula:

$$\text{f-meazure}=\frac{2*\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

Experimental system uses 10-fold cross validation method. The data was divided into 10 equal parts, taking 9 parts for training respectively and the other for testing, the results after 10 experiments were recorded and overall assessment is presented in Table 2.

Table 2. The results of the tests

Testing	Precision	Recall	F-meazure
1	71.90%	88.82%	79.47%
2	83.27%	88.31%	85.71%
3	83.48%	93.03%	88.00%
4	81.23%	87.50%	84.25%
5	85.83%	84.20%	85.01%
6	82.59%	94.53%	88.16%
7	79.69%	87.93%	83.61%
8	77.72%	84.03%	80.75%
9	82.08%	93.11%	87.25%
10	82.87%	88.85%	85.76%
Average	**81.07%**	**89.03%**	**84.80%**

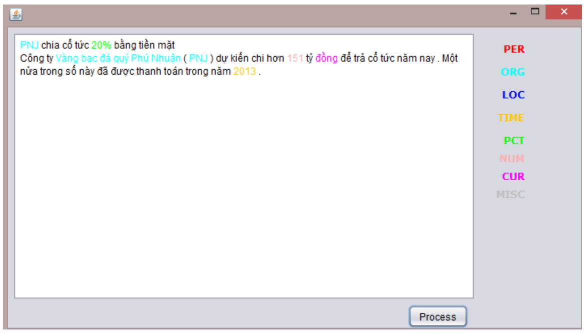
In addition, test results are also considered for each label type with the results in Table 3:

Table 3. Test results with labels

Entity name	Precision	Recall	F-measure
CUR	81.25%	81.25%	81.25%
LOC	59.09%	100.00%	74.29%
NUM	100.00%	99.08%	99.54%
ORG	52.94%	75.00%	62.07%
PCT	100.00%	91.30%	95.45%
PER	92.00%	92.00%	92.00%
TIME	67.44%	100.00%	80.56%

D. Application building

Based on the model that was built and tested, an application was built, applying CRF model to identify entities in Vietnamese text. With a text file input, the app analyzes text content, identifies identifier entities in the text, and changes colors for phrases that correspond to different labels. For example, identified entities with B-PER, I-PER labels change color to red, labels are B-LOC, I-LOC changes color to green, etc. shown in Figure 2.



PNJ chia cổ tức 20% bằng tiền mặt
Công ty Vàng bạc đá quý Phú Nhuận (PNJ) dự kiến chi hơn 151 tỷ đồng để trả cổ tức năm nay . Một nửa trong số này đã được thanh toán trong năm 2013 .

Fig. 2. Entity identification application

IV. CONCLUSION

The main result presented in the paper is an open source application system that allows the training to create an identifier entity identification model based on CRF model.It is included training, testing and application modules for Vietnamese text. The system f-measure reaches 84.8% on the test data set. With the procedure described in previous section, the system can receive different customized training data (for example, in different domains) depending on the needs to create suitable models. identification of identifier entity in Vietnamese documents.

To increase the accuracy of entity identification in the system, the training data source needs to be large and accurate. We will continue to explore and collect new sources of data and expand the types of entities that need to be identified, adding new laws to create attributes that support the training process to increase the accuracy of the paradigm.

REFERENCES

[1] Nancy Chinchor and Patty Robinson, MUC-7 Named Entity Task Definition, Proc. Sixth Messag. Underst. Conf. MUC6, p. 21, 1997.
[2] Alireza Mansouri, Lilly Suriani Affendey, and Ali Mamat, Named Entity Recognition Approaches, J. Comput. Sci., vol. 8, pp. 339–344, 2008.
[3] Sunita Sarawagi, Information Extraction, vol. 1, no. 3, pp. 261–377, 2008.
[4] Douglas E. Appelt, Jerry R. Hobbs, John Bear, and David Israel, SRI International FASTUS system MUC-6 test results and analysis, in MUC-6, NIST, 1995.
[5] Phil Blunsom, Hidden Markov Models, Lect. notes, 2004.
[6] A. McCallum, D. Freitag, and F. Pereia, Maximum entropy markov models for information extraction and segmentation, in International Conference on Machine Learning, 2000.
[7] John Lafferty, Andrew Mccallum, and FCN Fernando C. N. Pereira, Conditional Random Fields : Probabilistic Models for Segmenting and Labeling Sequence Data, in ICML '01 Proceedings of the Eighteenth International Conference on Machine Learning, 2001, vol. 2001, pp. 282–289.
[8] John M. Hammersley and Peter Clifford, Markov fields on finite graphs and lattices, 1971.