idrbt-team-a@IECSIL-FIRE-2018: Relation Categorization for Social Media News Text

N. Satya Krishna[2,3], S. Nagesh Bhattu[1], and D. V. L. N. Somayajulu[3]

[1] NIT Tadepalligudem, West Godavari District, Andhra Pradesh, India
nageshbhattu@nitandhra.ac.in
[2] IDRBT, Road No.1 Castle Hills, Masab Tank, Hyderabad, Telangana, India
satya.krishna.nunna@gmail.com
[3] NIT Warangal, Telangana, India
{soma}@nitw.ac.in

Abstract. This working note presents a statistical based classifier for text classification using entity relationship information present in the input text. We observed that parts-of-speech tags and named entities information will help us to predict the relationship between entities. We also presented the procedure for predicting POS tags and named entities, which we considered as the sources of information for entity relationship. These features (POS tags, NE) along with the words, in input text sentence, are used as input features to classify the given input into any one of the predefined relationship class. It also presents the experimental details and performance results of this classifier on five Indian language datasets such as *hindi, kannada, malayalam, tamil* and *telugu*.

Keywords: Relation Extraction· Parts-Of-Speech tagging· NER· Logistic regression· IR.

1 Introduction

Relation Extraction(RE) is an important subtask in Natural Language Processing(NLP) pipeline to convert the unstructured data format to structured data format by extracting the relationship information among the entities in the natural text. The main goal of a RE task is to identify and extract the relationships between two or more entities existing in the given unstructured data and classifies the given text based on the extracted relationship.

With the increase in the usage of internet, the unstructured digital data has been increasing exponentially in the form of blogs, research documents, posts, tweets, news articles, and question-answering forms. This unstructured digital data consists important information in the hidden form. The objective of Information Retrieval(IR) is to develop tools for extracting this information automatically. To extract this information, it requires the conversion of unstructured digital data into structured form by predicting the named entities and relationships existed among those entities in the unstructured data.

For example consider the sentence shown in table-1 along with its POS tags and named entities. The sentence has three entities *person, occupation*, and *organization*. We can extract these entities information using NER tools. With these entities information, RE algorithm identifies the existence of entities in the given text, but not the relationship information among those entities. In this example there are two relationships among the three entities. The First relationship is *woking as* between *john-person* and *assistant professor-occupation* entities. The second relationship is *working in* between *john-person* and *IITD-organization* entities.

Table 1. Eaxmple Sentence-1

sentence1	john	working	as	an	assistant	professor	in	IITD
POS tags	NNP	V_VM	PSP	DT	JJ	NN	PSP	NNP
Named Entities	Person	other	other	other	occupation	occupation	other	organization

Named entities in the second example, shown in table-2, are *person, location, event* and *datenum*. Its corresponding POS tags and words are the sources to find the relationship among these entities. In this example two relationships *discovered* and *discovered in* are exist. The *discovered* relationship between *Columbus-person* and *America-location*. second relationship *discovered in* between *America-location* and *1492-datenum*.

Table 2. Example sentence-2

sentence2	Columbus	discovered	America	in	1492
POS tags	NNP	V_VM	NNS	PSP	CD
Named Entities	Person	event	location	other	datenum

As shown in the above examples, our approach is to consider POS tags along with named entities, as they are good source for relationship extraction and text classification. Since, most of the times verbs, prepositions and verbs followed by preposition can provide the relationship information, in this work we built a machine learning based classifier to extract generic relationship pattern using POS tags and named entities along with words in a given training example in training phase. There are many applications in which we can use RE task. A bio-medical tool EDGAR, presented in [8], extracts the relationship information between the drugs and gems with cancer disease. It is implemented using NLP tools to generate POS tags. The recent survey[7], on Relation Extraction(RE) from unstructured text, presented the description of different types RE models implemented for different applications. It also presents the dataset details and working procedures of these models by categorizing them based on the type of classifiers.

RE is a pivotal sub task in several natural language understanding applications, such as question answering [4]. Initially RE task was formulated as an essential task of information extraction by seventh Message Understanding Conference (MUC-7) [3]. Miller et al. [5] proposed a generative model to convert the unstructured text to structured form by extracting the entity relationship information using parts of speech tags and named entity information. Ru et al. [9] proposed a convolutional neural network (CNN) model for relation classification using core dependency phrases as input to the model. Initially he computed these core dependency phrases using Jaccard similarity score between the relation phrases exist in the knowledge base and the dependency phrases exist between two entities.

2 Overview of the Approach

This section describes the overall approach we followed in this work to classify the sentences according to the relationship information that exists among the entities in that sentence. For implementation we divided the approach into two stages. In first stage, as shown in the figure-1, we extracted the list of features which carries the entity information and relationship information for each sentence in train and testset. In second stage we train a sentence level classifier and build the model by feeding training sentences along with features extracted in the previous stage. These features are parts of speech tags and named entities in an input sentence. Later we applied this trained classifier to classify test sentences with its POS features and entity features.

3 Experiments

We experimented on five Indian language data sets using different classifiers. This section describes the problem definition, details of feature extraction, classification and datasets details with experimental results on those datasets.

3.1 Problem statement

Given a large collection of sentences C and each sentence is represented with a sequence of words, such as $w_0, w_1, w_2, ...w_n$, which contains the set of entities and relation among those entities. Classify these sentences into any one of the class labels given in training data using entity relationship information existing in an input sentence.

3.2 POS tagging and NER

As shown in the example mentioned in introduction section, the adverb, verb, conjunction and preposition type of words in a sentence are the sources of information for extracting the relation among the existing entities. Hence, we

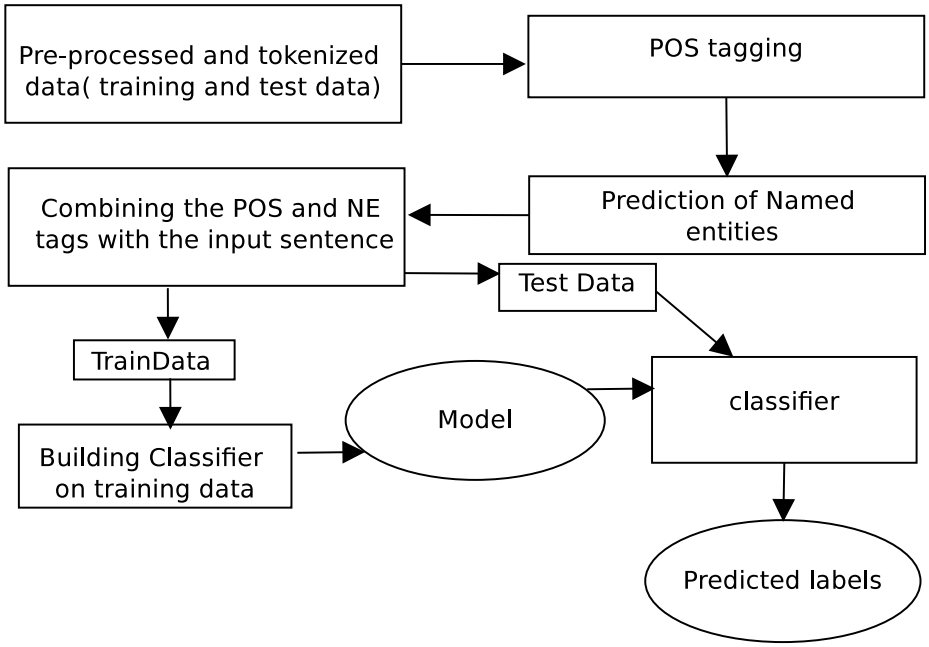

Fig. 1. Overview of approach

predicted the POS tag for each word in a sentence using different POS taggers for different languages. These POS tagging tools are freely available in IIIT-Hyd web site[4]. Though, they provided the POS taggers for these five Indian languages we utilized only *hindi, tamil* and *telugu* POS taggers. The details of these POS taggers and its source code is available in IIIT-Hyd website[5]. We predicted the named entities using a deep learning based Recurrent Neural Network called Bi-directional Long Short-Term Memory. The detailed description of this model is explained in our work submitted along with this paper in Arnekt-IECSIL@FIRE2018 shared-task workshop.

3.3 Relationship Extraction

As described in the previous section, we applied different types of machine learning based text classifiers to predict the entity relationship class labels. Initially we verified the performance of these classifiers on *telugu* language dataset. As logistic regression based classifier gave the highest accuracy(76.35%) compared to other classifiers NBEM(67.5%) and MLP(75.98%) we applied this classifier for the remaining datasets.

[4] http ltrc.iiit.ac.inshowfile.php?filename=downloadsshallow_parser.php
[5] http:ltrc.iiit.ac.indownload.php

Naive Bayes EM (NBEM) classifier: NBEM is a semi-supervised learning based classifier[6], which takes the test data along with the training data to utilize information existing in unlabeled data for predicting the class labels. The given training and test datasets are denoted with $D_l = (x_0, y_0)$,(x_1, y_1), (x_2, y_2), ... (x_l, y_l) and $D_u = (x_{l+0})$, (x_{l+1}), (x_{l+2}), ... (x_{l+u}) respectively. The NBEM classifier learns on dataset $D_n = D_l \bigcup D_u$ by maximizing the following objective function shown in equation-1. Here $n = l + u$ and l denotes the number of sentences in training data. u denotes the number of sentences in test data. x_j and y_j are denoting j^{th} input and its corresponding output label.

$$maximize \sum_{j=1}^{|D_l|} log(\phi(x_j, y_j)) + \sum_{k=1}^{|D_u|} log(\phi(x_k, \widehat{y}_k)) \tag{1}$$

It learns the model parameters by applying Expectation and Maximization(EM) algorithm. In EM algorithm it predicts the labels of each test sample in Expectation step(E-step) and updates the parameters in Maximization step(M-step) in each iteration.

Logistic Regression classifier: In logistic regression[10] we use discriminate probabilistic method to build the classifier. If we consider the training data as D_n = (x_0, y_0),(x_1, y_1), (x_2, y_2), ... (x_n, y_n) then this classifier learns the parameters by maximizing the following objective function shown in equation-2. Here, ϕ is a conditional probability for occurrence of label y_j given input observation x_j. We compute this probability using softmax function as shown in equation-3. In this function L denotes the set of output labels in the given dataset. θ_{y_l} is a parameter vector of size equal to the length of the input.

$$maximize \sum_{j=1}^{|D_n|} log(\phi(y_j/x_j)) \tag{2}$$

$$\phi(y_j/x_j) = \frac{\exp(\theta_{y_j}^T.x_j)}{\sum_{l=1}^{|L|} \theta_{y_l}^T.x_j} \tag{3}$$

Multilayer Perceptron: It is a special type of feedforward neural network in which it uses a non-linear activation function in each neuron of hidden layer and output layer. We used the back propagation algorithm while learning the MLP classifier on training data. In training phase, it back propagates the error occurred for each input observation. While back propagating the error, it adjust the weights by minimizing the following objective function shown in equation-4, where $E_i(x_n)$ gives the error at i^{th} output node for the n^{th} input observation. We applied the gradient descent algorithm to change the weights.

$$E_i(x_n) = \frac{1}{2} \sum_n (y_i^n - \widehat{y_i}^n)^2 \tag{4}$$

3.4 Dataset

Arnekt-IECSIL@FIRE2018 shared task [2] provides five Indian language datasets [1] such as *hindi, kannada, malayalam, tamil* and *telugu*. Each language dataset has three different files with text in its corresponding language script. Among these three files, one is training file and other two are test-1 and test-2 files. All these files have data in the form of one input sequence per line. Except test-1 and test-2, the training data file contains labels for each input sequence. Each test file has 20% of data from the overall dataset. The training file in each dataset has different number of distinct labels. The number of class labels are 16, 14, 13, 17 and 14 in *hindi, kannada, malayalam, tamil* and *telugu* training data respectively. The table-3 describes the details regarding the number of sentences, number of words, number of unique words in each file of all datasets.

Table 3. Corpus description

Dataset name	FileType	# Sentences	# words	Vocabulary Size
Hindi	Train	56775	1134673	75808
	Test-1	18925	380574	37770
	Test-2	18926	374433	37218
Kannada	Train	6637	123104	35961
	Test-1	2213	40013	15859
	Test-2	2213	40130	15862
Malayalam	Train	28287	439026	85043
	Test-1	2492	147259	39319
	Test-2	2492	144712	38826
Tamil	Train	64833	882996	130501
	Test-1	21611	294115	61754
	Test-2	21612	292308	61366
Telug	Train	37039	494500	76697
	Test-1	12347	163343	34984
	Test-2	12347	163074	35092

3.5 Results

Evaluation of this model is done based on the following metrics as specified in the shared task guide lines.

$$Accuracy = \frac{No.Of\ sentences\ are\ assigned\ with\ the\ correct\ label}{No.Of\ sentences\ in\ the\ dataset} \tag{5}$$

$$Precision(P_i) = \frac{No.Of\ sentences\ are\ correctly\ labeled\ with\ label_i}{No.Of\ sentences\ are\ labeled\ with\ label_i} \tag{6}$$

$$Recall(R_i) = \frac{No.Of\ sentences\ are\ correctly\ labeled\ with\ label_i}{Total\ No.Of\ sentences\ with\ label_i\ intest\ data} \tag{7}$$

$$fscore(F_i) = \frac{2 * P_i * R_i}{P_i + R_i} \tag{8}$$

$$Overall\ fscore(F) = \frac{1}{|L|} * \sum_{iinL} F_i \tag{9}$$

The results in table-4 and 5 summarizes the accuracy and F-Scores of the model on testset-1 and 2 of five languages. In Per-Evaluation and Final-Evaluation this model has high accuracy for *kannada* dataset compared with other models presented in this competition. For the remaining four languages this model is in the second position. This same order is repeated in F-Score performance of this model.

Table 4. Accuracy of the model in Pre-Evaluation and Final-Evaluation

	Hindi	Kannada	Malayalam	Tamil	Telugu
Pre-Evaluation(Testset-1)	79.92	57.98	59.43	78.43	76.35
Final-Evaluation(Testset-2)	79.21	57.34	57.86	78.44	76.13

Table 5. F1-Scores of the model in Final-Evaluation

	Hindi	Kannada	Malayalam	Tamil	Telugu
Final-Evaluation(Testset-2)	31.64	33.5	28.31	51.59	45.86

According to our analysis, the reason for showing less F-Score by this model is the error in prediction of POS tags and named entities, made this model to miss classify the given sentences.

4 Conclusion

We presented an approach which makes use of POS tag information for relation categorisation. Our approach was general and applicable for all the languages used in the shared-task. We could not get the POS tags of malayalam and kannada which can be further improved with such additional information which is crucial for the success in the shared task.

References

1. Barathi Ganesh, H.B., Soman, K.P., Reshma, U., Mandar, K., Prachi, M., Gouri, K., Anitha, K., Anand Kumar, M.: Information extraction for conversational systems in indian languages - arnekt iecsil. In: Forum for Information Retrieval Evaluation (2018)

2. Barathi Ganesh, H.B., Soman, K.P., Reshma, U., Mandar, K., Prachi, M., Gouri, K., Anitha, K., Anand Kumar, M.: Overview of arnekt iecsil at fire-2018 track on information extraction for conversational systems in indian languages. In: FIRE (Working Notes) (2018)
3. Chinchor, N.: Proceedings of the 7th message understanding conference. Columbia, MD: Science Applications International Corporation (SAIC) (1998)
4. Hazrina, S., Sharef, N.M., Ibrahim, H., Murad, M.A.A., Noah, S.A.M.: Review on the advancements of disambiguation in semantic question answering system. Information Processing & Management **53**(1), 52–69 (2017)
5. Miller, S., Fox, H., Ramshaw, L., Weischedel, R.: A novel use of statistical parsing to extract information from text. In: Proceedings of the 1st North American Chapter of the Association for Computational Linguistics Conference. pp. 226–233. NAACL 2000, Association for Computational Linguistics, Stroudsburg, PA, USA (2000), http://dl.acm.org/citation.cfm?id=974305.974335
6. Nigam, K., McCallum, A.K., Thrun, S., Mitchell, T.: Text classification from labeled and unlabeled documents using em. Machine learning **39**(2-3), 103–134 (2000)
7. Pawar, S., Palshikar, G.K., Bhattacharyya, P.: Relation extraction: A survey. arXiv preprint arXiv:1712.05191 (2017)
8. Rindflesch, T.C., Tanabe, L., Weinstein, J.N., Hunter, L.: Edgar: extraction of drugs, genes and relations from the biomedical literature. In: Biocomputing 2000, pp. 517–528. World Scientific (1999)
9. Ru, C., Tang, J., Li, S., Xie, S., Wang, T.: Using semantic similarity to reduce wrong labels in distant supervision for relation extraction. Information Processing & Management **54**(4), 593–608 (2018)
10. Walker, S.H., Duncan, D.B.: Estimation of the probability of an event as a function of several independent variables. Biometrika **54**(1-2), 167–179 (1967)