DOI 10.29042/2017-2025-2028

*Helix Vol. 7(5): 2025-2028*

# USING LARGE SCALE DEEP LEARNING METHOD TO PREDICT PUNCTUATIONS IN TELUGU LANGUAGE

**[1]Ch.Prathima, [2] Ch.Sreenu Babu, [3] Naresh Babu Muppalaneni, *[4] Kishore C**

[1, 2 3, 4]Sree Vidyanikethan Engineering College, Tirupati, India

*Email: mr.c.kishore@gmail.com*

**Received:8th June 2017, Accepted: 25th June 2017, Published: 1st September 2017**

**Abstract**

Punctuation performs an important role in language processing. However, automated speech recognization systems only output plain terms sequences. It really is then appealing to predict punctuations on simple word sequences. Earlier works are focused on using lexical features or prosodic cues captured from small corpus to predictable simple punctuations. When compared with simple punctuations, rich punctuation provides more meaningful information and are more challenging to predict. In this paper, LSDL model is suggested to predict rich punctuations on large-scale corpora. Experiments are performed on both in-domain and out-of-domain datasets for prediction of punctuations. The result of the Experiments shown that LSDL can significantly outperform the original CRF-based model. Furthermore, large-scale corpora are demonstrated to bring large improvement, and presenting POS tags and Chunking information in LSDL model on small corpus to improve performance.

**Keywords:** LSDL, Deep Learning, Punctuation Prediction

## 1. INTRODUCTION

Nowadays, with the fast development of IT, innumerable levels of information have been created and disseminated, a huge part which is speech information. The most frequent way to investigate speech data is to convert them into text message so that natural language processing techniques, such as analysis of sentiments, extraction of information and machine translation, can be applied. Research [2] has proven that punctuations are essential for these downstream processing. However outputs of the almost all of automated speech reorganization (ASR) systems [1] simply consist of streams of words.

There were some research on this problem, specifically, punctuation prediction or punctuation recovery. Most of earlier works rely on lexical features or prosodic cues [2]. In such cases, supervised learning techniques are used, but there is no large and high-quality corpus for training such models, especially in Telugu language. Most of the research works focuses on small corpora such as PTB and CTB (Telugu Tree Bank), and manual speech transcription. Due to commercial and cost factors large-scale manual transcriptions are unavailable for general public. And the public corpora are usually small and only cover few categories, as the abundance and variety of training data are of significant importance for the punctuation prediction model to assure higher accuracy and reliability as well as generalization ability.

Text materials can be split into two types: formal text materials and informal text materials, according to their writing styles. Media is a typical genre of formal text, which generally has standard punctuations and structure. While a more substantial part of a text on the web is informal text, such as posts and microblogs (Twitter and Weibo). Commonly, speech transcriptions are much closer to informal text because colloquialism contains words and phrases that are being used in ordinary discussion, including slangs, idioms and abbreviations.

In this paper, we focus on rich Telugu language punctuations prediction. To resolve the situation triggered by restricted training data and high cost of manual labeling, we use a huge scale corpus gathered from various resources including Wikipeida, News, Weibo and real speech transcriptions. In our experiments, we shown that large-scale corpora bring large improvement than small corpora.

Punctuation prediction is generally considered as an example of sequence labeling tasks, treating punctuations as labels of words. Currently, recurrent neural networks(RNNs), especially long short-term memory RNN, have been a dominating approach for sequence labeling. But there are few works implementing RNNs to punctuation prediction except from [3]. They bring a two-stage Large Scale Deep Learning (LSDL) model to revive punctuation in speech transcriptions using both lexical features and prosodic cues, reducing the error by at most 16% in comparison to 4- gram+DT-p method. It shows the promising power of deep learning methods. To avoid manual work, we simply used lexical features as a source of LSDL. [4] Demonstrates multi-view learning framework can offer similar performance. We then add POS tags and chunking information to our model with multi-view learning framework. By introducing additional information, our LSDL model makes improvement on small corpora.

## 2. METHODS

### 2.1. Task Formulation

Telugu language punctuation prediction is an activity that inserts punctuation symbols properly to a unpunctuated Telugu word sequence. In our task, this technique is conducted as a sequence of labeling task with five tags. Here we consider four common punctuations: comma, period, exclamation mark and question mark. Each word has its own label as per the given table 1. 'None' means no punctuation behind the word.

**Table 1: Kinds of Punctuations**

| Punctuation after a word | Symbol | Tag |
|---|---|---|
| Comma | , | CO |
| Period | . | PR |
| Exclamation Mark | ! | EM |
| Question Mark | ? | QM |
| None | | NO |
| Sentence Boundary | | SB |

For instance, a punctuated sentence is first segmented into word sequence and then mounted on its punctuation labels.

పొగాకు, పత్తి, రైస్, చెరుకుగడ ఆంధ్రప్రదేశ్ ప్రధాన పంటలు. (Tobacco, cotton, Rice and sugarcane are the main crops in Andhra Pradesh.)
Labeled sequence words

పొగాకు/CO పత్తి/CO రైస్/CO చెరుకుగడ ఆంధ్రప్రదేశ్ ప్రధాన పంటలు /PR

In practical system, the metrics precision and recall are used to evaluate the performance, we use precision (denoted by P), recall (denoted by R) and F1-score (the harmonic mean of precision and recall, denoted by F) for scoring. Besides scoring four kinds of punctuation respectively, we also evaluate the performance of sentence boundary by not considereing the differences between various punctuations, concentrating on whether to place a punctuation (denoted by SB).

### 2.2. CRF Based Model

Conditional Random Fields (CRFs) [5] has been universally applied for punctuation prediction, since it models the conditional distribution of whole observed sequences which relaxes the impartial assumption of Hidden Markov Models (HMMs). Also, CRFs allow us to include more observation features to labeling.

We use a CRF toolkit CRF++, a customizable, simple and open source development of CRFs for segmenting/labeling sequential data, to develop our CRF-based model. For an input sequence $u = (u_1, u_2, ..., u_T)$, where $u_i$ is the $i^{th}$ feature in the sequence with size $T$ and an output sequence $v = (v_1, v_2, ..., v_T)$, the global feature vector (GFV) is calculated by

$$f(v,u) = \sum_i f(v,x,i)$$

The target for training is to increase the conditional probability is given by

$$P_\lambda(v|u) = \frac{\exp(\lambda \,.\, f(v,u))}{Z_\lambda(u)}$$

where $Z_\lambda(u)$ is a normalization factor that ensures that the summation of the likelihood of all sequences of outputs is one, given by

$$Z_\lambda(u) = \sum_v \exp(\lambda \,.\, f(v,u))$$

The proposed CRF model, we exploit both words and POS tags in the framework with 9 features. Additional information about CRF feature are shown in table 2, where $w_i$ represents the existing word and $p_i$ represents the respective POS tag.

**Table 2: CRF Features**

| No. | Feature | Description |
|---|---|---|
| W0 | $W_{i-4}$ | The fourth word before the current word. |
| … | … | … |
| W8 | $W_{i+4}$ | The fourth word after the current word |
| P01 | $P_{i-4}P_{i-3}$ | The fourth and third POS tags before the current POS tags |
| … | … | … |
| P67 | $P_{i+3}P_{i+4}$ | The fourth and third POS tags after the current POS tags |
| P012 | $P_{i-4}P_{i-3}P_{i-2}$ | The fourth, third and second POS tags before the current POS tags |
| … | … | … |
| P567 | $P_{i+2}P_{i+3}P_{i+4}$ | The fourth, third and second POS tags after the current POS tags |

With the results of CRF, our model provides a secondpass operation, employing handicraft guidelines to improve the performance on special conditions.

The guidelines are too trivial describe, so here we group them into four main types:

- Conjunction: place comma before a conjunction.
- Parenthesis: place comma or period to leading and rear of parenthesis.
- Interrogative sentence: place a question mark after the tone word if interrogative sentence is recognized.

 

- Exclamatory mark: place exclamation mark following the tone word if exclamatory mark is detected.

### 2.3. Large-Scale Deep Learning Model

Recurrent Neural Networks is an extension of neural network. Difference with the traditional neural networks is that the current hidden state of RNN would depend on that of previous time. It is an intuitive to employ RNN to sequence labeling tasks, because the prior words play an important role in understanding the framework. Though RNN has achieved great success in many sequence processing tasks, it is suffering from gradient vanishing and gradient explosion [6].

LSDL extends Recurrent Neural Networks by presenting a memory cell c with the control gates (input i, result o and forget gate f) at each and every step. These gates control the behaviors of memory cells. At each step t, the memory cell state depends upon the input $x_t$, previous hidden state $h_{t-1}$ and previous cell state $c_{t-1}$.

## 3. EXPERIMENTS

### 3.1. Data and Experimental Results

Experiments are performed on two different corpora: informal corpus and formal corpus. Each corpus includes three subsets: the large one which is treated as traning set and the remaining are "out-of-domain" training set and "in-domain" tranining set. A testing set that is extracted from the same source as training set is categorized as "in-domain", conversely "out-of-domain". It is Observed that "in-domain" testing set and training set are sharing the same source, there is no overlap between them. A list of all datasets is shown in a table 3.

**Table 3: Datasets**

| Datasets | | Contents | Size(Byte) |
|---|---|---|---|
| Informal Corpus | Training | Weibo | 1.5G |
| | In-domain | Weibo | 4.5M |
| | Out-of-domain | Speech Transcriptions | 1.2M |
| Formal Corpus | Training | Sina News, Wikipedia | 1.4G |
| | In-domain | Sina News, Wikipedia | 7.5M |
| | Out-of-domain | People's Daily, Articles | 12M |

In this paper, we present 3 sets of experiments in terms of learning algorithm, data scale and LSDL. As discussed in section 2.1, we evaluate the performance of four types of punctuations as well as the sentence boundary (denoted by SB).

We sample 1/10, 1/100, 1/1000 instances from full training data as new training sets, to explore the way the level of performance is impacted on training data sets.

Before going to perform experiments, we have to normalize the data by removing the hyper-links, tags and other irrelevant characters, unify Telugu language characters(i.e. transforming Traditional Telugu language to Simplified Telugu language), and map all punctuations to four types: comma(,), period(.), exclamation mark(!), question mark(?). The normalization of punctuations is shown in table 4

**Table 4: Normalized Punctuations**

| Original Punctuations | Mapped Punctuations |
|---|---|
| , 、 ; ; —— : :“ ” ‘ ’ | , |
| . . ... ~ | . |
| ! ! | ! |
| ? ? | ? |

### 3.2. Linguistic Features

Two categories of linguistic features are used in this model: part-of-speech tags and chunking. We use Stanford POS tagger [7] for assigning part-of-speech tag to each and every word. The part-of-speech tags follows Penn Treebank format. To use sentence level information, we can also used Stanford Constituent Parser [8] to parse word stream and then convert the parsing tree to chunking labels [9, 10].

## 4. RESULTS

### 4.1. CRF Model vs LSDL Model

The result that was given by CRF-based model and LSDL model are shown in table 5. LSDL model overwhelmingly surpasses CRFbased model on all punctuations. Especially, LSDL model makes great improvement on sentence boundary detection. Long-term dependency may be the main reason behind it. In Telugu language, a word stream can be express in different tones without the reformation. For instance: ఇది నాది (This is mine) can be declarative (ఇది నాది .), interrogative (ఇది నాది ? ) and exclamatory (ఇది నాది!). These shades rely upon speaker's emotion, which is often revealed by larger context. Also, in Telugu grammar a sentence can have several predicate, rendering it difficult to differentiate comma and period.

**Table 5: The F1 score on Rich Punctuations on Different Datasets**

| Training | Testing | Label | CRF Model | LSDL Model |
|---|---|---|---|---|
| Formal Corpus | In-domain | CO | 47.43 | **66.15** |
| | | PR | 55.78 | **66.63** |
| | | EM | 43.03 | **52.98** |
| | | QM | 7.91 | 2.79 |
| | | SB | 55.99 | **78.49** |
| | Out-of-domain | CO | 47.75 | **66.60** |
| | | PR | 60.33 | **68.05** |
| | | EM | 45.51 | **52.58** |

| | | | | |
|---|---|---|---|---|
| | | QM | 4.95 | **11.41** |
| | | SB | 56.98 | **77.74** |
| Informal Corpus | In-domain | CO | 54.96 | **69.57** |
| | | PR | 37.76 | **58.78** |
| | | EM | 53.63 | **57.36** |
| | | QM | **67.66** | 62.97 |
| | | SB | 59.58 | **83.42** |
| | Out-of-domain | CO | 30.52 | **48.11** |
| | | PR | 37.07 | **52.62** |
| | | EM | 49.02 | **61.66** |
| | | QM | 7.09 | **32.99** |
| | | SB | 41.87 | **72.02** |

### 4.2. Scale vs Performance

The result shown in table 6 confirms that larger-scale corpora bring better performance and generalization capability. with the increasing level of corpus, the performance gain that larger corpus brings is decreasing. one of the reasons is the limitation of model scale. When we raise the hidden size of LSDL model to 500, the performance on full formal training data ascends for 78.49 to 79.96.

**Table 6: The F1-Score of sentence boundary detection using different scales of training sets**

| Corpus | Testing | 1/1000 | 1/100 | 1/10 | Full |
|---|---|---|---|---|---|
| Formal Corpus | In-domain | 51.98 | 65.71 | 77.32 | 78.49 |
| | Out-of-domain | 50.79 | 65.2 | 76.58 | 77.74 |
| Informal Corpus | In-domain | 60.94 | 75.14 | 81.74 | 83.42 |
| | Out-of-domain | 51.12 | 63.76 | 68.83 | 72.02 |

## 5. CONCLUSION

Our large-scale deep learning model achieves overwhelmingly improvement in rich punctuations prediction than traditional method. We are concluding that LSDL model has a better ability to predict the sentence boundary, but because of characteristics of Telugu language punctuation, there continues to be a issue of determining which punctuation to place at the boundary. Our work also shows that large-scale corpus helps promote the performance and generalization capability in both formal and informal corpora.

## REFERENCES


[1] Xueyang Wu, Su Zhu, Yue Wu, Kai Yu, "Rich Punctuations Prediction Using Large-scale Deep Learning", Chinese Spoken Language Processing (ISCSLP), 2016 10th International Symposium on, 2016.

[2] B. Favre, R. Grishman, D. Hillard, H. Ji, D. Hakkani-Tur, and M. Ostendorf, "Punctuating speech for information extraction," ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings, 2008.

[3] Stolcke, E. Shriberg, and M. Harper, "Using Conditional Random Fields For Sentence Boundary Detection In Speech," 2005.

[4] O. Tilk and T. Alumae, "LSTM for punctuation restoration in ¨ speech transcripts," Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH, Vol:54. 2015.

[5] P. Dhillon, D. P. Foster, and L. H. Ungar, "Multi-view learning of word embeddings via cca," in Advances in Neural Information Processing Systems 24, J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett, F. Pereira, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2011.

[6] J. Lafferty, A. McCallum, and F. C. N. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," ICML '01 Proceedings of the Eighteenth International Conference on Machine Learning, Vol: 8, 2001.

[7] S. Hochreiter, S. Hochreiter, J. Schmidhuber, and J. Schmidhuber, "Long short-term memory." Neural computation, vol. 9, no. 8, 1997.

[8] C. Xu, D. Tao, and C. Xu, "A Survey on Multi-view Learning," Cvpr, vol. 36, no. 8, 2015. [Online]. Available: http://arxiv.org/abs/1304.5634.

[9] M. Zhu, Y. Zhang, W. Chen, M. Zhang, and J. Zhu, "Fast and Accurate Shift-Reduce Constituent Parsing," Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics,Vol: 1, 2013.

[10] E. F. T. K. Sang and S. Buchholz, "Introduction to the CoNLL- 2000 shared task: Chunking," in Proceedings of CoNLL, Vol: 0, 2000.


 