

# A Chinese Dataset with Negative Full Forms for General Abbreviation Prediction

Yi Zhang, Xu Sun

MOE Key Laboratory of Computational Linguistics, Peking University  
School of Electronics Engineering and Computer Science, Peking University  
zhangyi16, xusun@pku.edu.cn

## Abstract

Abbreviation is a common phenomenon across languages, especially in Chinese. In most cases, if an expression can be abbreviated, its abbreviation is used more often than its fully expanded forms, since people tend to convey information in a most concise way. For various language processing tasks, abbreviation is an obstacle to improving the performance, as the textual form of an abbreviation does not express useful information, unless it's expanded to the full form. Abbreviation prediction means associating the fully expanded forms with their abbreviations. However, due to the deficiency in the abbreviation corpora, such a task is limited in current studies, especially considering general abbreviation prediction should also include those full form expressions that do not have general abbreviations, namely the negative full forms (NFFs). Corpora incorporating negative full forms for general abbreviation prediction are few in number. In order to promote the research in this area, we build a dataset for general Chinese abbreviation prediction, which needs a few preprocessing steps, and evaluate several different models on the built dataset. The dataset is available at <https://github.com/lancopku/Chinese-abbreviation-dataset>.

**Keywords:** Chinese abbreviation, negative full forms, conditional random field, long-short term memory

## 1. Introduction

Abbreviation processing mainly consists of three tasks, that is, abbreviation expansion, abbreviation recognition, and abbreviation prediction. Expanding the short form of an expression to its full form is called abbreviation expansion. Extracting the short form and full form pairs from the context is called abbreviation recognition. Abbreviation prediction refers to predicting the short form of an expression according to its full form. In this paper, we focus on the last task, i.e., abbreviation prediction. Abbreviation prediction plays an important role in various language processing tasks, because accurate abbreviation prediction will help improve performance. Sun et al. (2009) shows that better abbreviation prediction will help improve the performance of abbreviation recognition. Abbreviation prediction also benefits other tasks. For example, in an information retrieval (IR) system, a large number of the web pages contain only abbreviations. It will be helpful if we can estimate abbreviations of a query, because successful abbreviation prediction may improve the recall of IR systems as Sun et al. (2013a) showed. In addition, Yang et al. (2012) showed that Chinese abbreviation prediction can improve voice-based search quality.

English abbreviations are usually formed as acronyms. Studies for English abbreviation proposed various heuristics for abbreviation prediction (Park and Byrd, 2001; Wren et al., 2002; Schwartz and Hearst, 2002). For example, use of initials, capital letters, syllable boundaries, stop words, etc. These studies performed well for English abbreviations. While Chinese abbreviations are quite different from English ones. Yang et al. (2012) showed that Chinese abbreviations are usually generated by three methods, reduction, elimination, and generalization. Characters are selected from the expanded full name to form the abbreviation. However, there are no general rules to convert a complete term into an abbreviation. As shown in Figure 1, an abbreviation may be generated using the first character and the last character. Sometimes, characters in the middle can be included while the last abbreviation takes the first two characters of the words. However, it is not necessary for Chinese abbreviations to take the first characters of words. They frequently take non-initial characters, like the last example in Figure 1. Chinese abbreviations are derived via a customary lexical process. Native speakers may associate a fully expanded term with its abbreviation by some intuition. But the process can not be adequately explained by any linguistic theory: Chang and Lai (2004) and Chang and Teng (2006).

Besides the irregularity of abbreviating phrases and terms, another main problem is caused by negative full forms. A word annotated with a negative full form means the word has no abbreviation at all. We usually recognize abbreviations or make abbreviation predictions in the text. Unfortunately, NFFs take up a large portion of Chinese words or phrases in the real world. With the strong noise, distinguishing the full forms with valid abbreviations is more difficult. This undoubtedly increases the difficulty of abbreviation prediction.

Many approaches have been proposed in the post studies. Sun et al. (2008) employed Support Vector Regression (SVR) for scoring abbreviation candidates. This method

珠穆朗玛峰 (Mount Qomolangma)	→	珠峰
奥林匹克运动会 (Olympic Games)	→	奥运会
北京大学 (Peking University)	→	北大
清华大学 (Tsinghua University)	→	清华
黄金市场 (gold market)	→	金市

Figure 1: Different cases of generating abbreviations

outperforms the hidden Markov model (HMM) in abbreviation prediction. Yang et al. (2009) proposed to formulate abbreviation generation as a character tagging problem and the conditional random field (CRF) then can be used as the tagging model. Sun et al. (2009) combined latent variable model and global information to predict abbreviations. Zhang et al. (2016) used a recurrent neural networks to predict abbreviations for Chinese named entities.

However, most studies of abbreviation prediction focus on positive full form, which means a word has a valid abbreviation. Apparently, this implicit lab assumption is not practical. Nonetheless, we barely see studies that consider NFFs. One of the main reasons is the shortage of abbreviation prediction data with NFFs, which is one of the main issues this work tries to solve.

Apart from the annotation of a dataset with NFFs, we also conduct a few preprocessing steps to facilitate the usage of the dataset. Chinese does not insert spaces between words or word forms after morphological changes. Hence, most of the Chinese natural language processing methods assume a Chinese word segmenter is used in a preprocessing step to produce word-segmented Chinese sentences as inputs. There is no exception for abbreviation prediction. Given original texts, we should first recognize the boundaries of words. After segmentation, we annotate the part-of-speech information of phrases and terms, because the part-of-speech information can serve as features to help make abbreviation prediction.

This paper details how the dataset is created and evaluates some frequently used models on the abbreviation prediction task.

## 2. Dataset

### 2.1. Considerations

**Commonness** Our intention is to build a dataset with NFFs, so it can be widely used for general abbreviation prediction. This requires that dataset contains most frequently-used full forms regardless of whether or not the form has valid abbreviations. The data sources should be reliable and accredited. Thus, we extract long phrases and terms in popular Chinese natural language processing corpora, which include People’s Daily corpora and SIGHAN word segmentation corpora.

**Usability** We also provide assisting information that is helpful for the abbreviation task in our dataset. Most existing methods treat abbreviation prediction as a sequence labeling problem. To make better tag predictions of characters, we usually need to extract some features. The word segmentation information and part-of-speech information are the most commonly used features. Unlike English, the smallest Chinese unit is a character rather than a word. There are no explicit boundaries between Chinese words. Since a full form usually can be segmented into several words and abbreviations often take characters from these words, segmentation information is most useful for abbreviation prediction. Another annotation is part-of-speech information. Many language processing tasks take part-of-speech information as features, including abbreviation prediction. The choice of characters which are used to form

abbreviations may be related to their part-of-speech information. For example, if a full form composed of an adjective and a noun has an abbreviation, the abbreviation usually takes a character of the adjective and a character of the noun. The same issue applies to a full form composed of an adverb and a verb when the full form has an abbreviation.

### Representativeness

Heilongjiang Province, Jilin Province, Liaoning Province  
东三省：黑龙江省，吉林省，辽宁省

Figure 2: A special case for abbreviation.

As the dataset should be representative of the common construction of abbreviations, we do not include special and irregular abbreviations, which include words outside the full form. As shown in Figure 2, “东三省” represents three provinces of China. It is a special type of abbreviations, since an abbreviation could represent several different terms. Without some background knowledge, what the abbreviation stands for can not be understood. Sometimes the characters of the abbreviation are not taken from original characters of the full form and the sequence labeling method is no longer applicable for this case. This kind of “abbreviation” is more like a general name for some terms. We do not include these special abbreviations in our dataset.

### 2.2. Data Source

Our text is from People’s Daily corpora and SIGHAN word segmentation corpora. We extract the long phrases and terms in the text. Then we classify the collected phrases and terms into two forms. One is the positive full form, which means the phrase or term has a valid abbreviation. Then its abbreviation is annotated. The other is the negative full form, which means the phrase or term does not have a general short form. Their abbreviations are “NULL”. Samples of the data are shown in Figure 3.

As mentioned before, we annotate word segmentation information and part-of-speech information for every phrase or term. Word segmentation is a fundamental task in Chinese processing. Many practical Chinese processing applications rely on Chinese word segmentation. Part-of-speech information is often used as features for further prediction. Most methods formulate these tasks as a sequence labeling problem. Various models achieved good performance on these tasks and some open source tools have been published for use. We used ICTCLAS, one of the best Chinese Lexical analyzers, to label the segmentation and part-of-speech information.

### 2.3. Statistics

We build a dataset that is made up of phrases and terms. There are 10,786 full forms in this dataset, including 8,015 positive full forms and 2,661 negative full forms. The phrases contain noun phrases, verb phrases, organization names, location names, and so on. The distribution is shown in Table 2. For experiments, we randomly sampled 7,551 samples as the training set, 1,078 samples as the development set and 2,157 samples as the testing set. We calculate the number of the words and characters (including

Positive / Negative Full Forms	Abbreviation
中国科学院 (Chinese Academy of Sciences)	中科院
二氯化碳酰 (Phosgene)	NULL
澳大利亚网球公开赛 (Australia Open Tennis Championships)	澳网
新加坡航空公司 (Singapore Airlines)	新航
独生子女家庭 (One-child family)	NULL
第一次世界大战 (The First World War)	一战
冠状动脉粥样硬化心脏病 (Coronary atherosclerotic heart disease)	冠心病
办公自动化 (office automation)	NULL

Figure 3: Samples of the collected data with NFFs. The “NULL” means no valid abbreviation.

Full Forms	
total entries	10,786
NFFs	2,661
total words	30,100
distinct words	8,293
total characters	60,877
distinct characters	2,557
average word length	5.644
Abbreviations	
total characters	23,077
distinct characters	1,687
average abbreviation length	2.140

Table 1: The statistics of the data. The results are count for full forms and abbreviations separately.

duplicates) in the data. We also count the number of distinct words and distinct characters. Then total characters of full forms divided by total entries is the average full form length. The average abbreviation length can be calculated in a similar way.

Category	Portion(%)
Noun Phrase	52.01%
Organization Name	26.84%
Verb Phrase	13.72%
Location Name	5.28%
Person Name	0.32%
Others	1.80%

Table 2: Distribution of the full forms in the data.

### 3. Models

#### 3.1. CRF

Tsuruoka et al. (2005) formalized the process of abbreviation prediction as a sequence labeling problem. Each

P: Produce the current character  
S: Skip the current character

(Beijing Office)

驻北京办事处

P S P P S S → 驻京办

Figure 4: Chinese abbreviation generation as a sequential labeling problem.

character in the expanded form is tagged with a label,  $y \in \{P, S\}$ , where the label  $P$  produces the current character and the label  $S$  skips the current character. In Figure 4, the abbreviation is generated using the first character, skipping the flowing character and then using the subsequent two characters. Because our task is general abbreviation prediction, we add another label “ $N$ ” to the tag set to label the characters in negative full forms. A number of recent studies have investigated the use of machine learning techniques. Traditional models like MEMM, perceptron and conditional random field perform well on such sequence labeling tasks. We use the well-known conditional random field (CRF) proposed by Lafferty et al. (2001) for sequential labeling.

We use features as follows:

- character feature : Input characters  $x_{i-1}$ ,  $x_i$  and  $x_{i+1}$
- character bi-gram : The character bigrams starting at  $(i-2) \dots i$ .
- Numeral: Whether or not the  $x_i$  is a numeral.
- Organization name suffix: Whether or not the  $x_i$  is a suffix of traditional Chinese organization names.
- Location name suffix: Whether or not the  $x_i$  is a suffix of traditional Chinese location names.
- Word segmentation information: After the word segmentation step, whether or not the  $x_i$  is the beginning character of a word.
- Part-of-speech information: The part-of-speech tag information of  $x_i$ .

In our abbreviation prediction task, the input sequence  $\mathbf{x}$  represents characters of a full form and output sequence  $\mathbf{y}$  represents symbolic labels based on abbreviations. The probability is defined as follows:

$$P(\mathbf{y}|\mathbf{x}, \mathbf{w}) = \frac{\exp[\mathbf{w}^T \mathbf{f}(\mathbf{y}, \mathbf{x})]}{\sum_{\mathbf{y}'} \exp[\mathbf{w}^T \mathbf{f}(\mathbf{y}', \mathbf{x})]} \quad (1)$$

where  $\mathbf{w}$  is the weight vector and  $\mathbf{f}$  is the mapping function.

Given a training set that consists of  $n$  labeled sequences  $(\mathbf{x}^i, \mathbf{y}^i)$  for  $i = 1 \dots n$ , the objective function is:

$$L(\mathbf{w}) = \sum_{i=1}^n \log P(\mathbf{y}^i | \mathbf{x}^i, \mathbf{w}) - R(\mathbf{w}) \quad (2)$$

where the second term is the  $L_2$  regularizer.

Method	Discriminate Acc(%)	All-Acc(%)	Char-Acc(%)
Heuristic System	73.20	25.77	65.79
Perc	87.48	54.89	87.02
MEMM	86.97	50.16	85.92
CRF-ADF	87.80	56.69	87.20
BLSTM	91.47	56.98	81.73

Table 3: Results on comparing different methods on generalized abbreviations.

### 3.2. BLSTM

As mentioned above, traditional methods depend heavily on features which need to be designed elaborately. Nowadays, more and more research focuses on neural networks, such as recurrent neural network (RNN), convolutional neural network (CNN) and some variants of RNN. These neural network models can extract features automatically. In natural language processing, traditional RNNs usually take the previous state  $h_{t-1}$  and the embedding  $x_t$  as the  $t$ -th input to calculate current state  $h_t$ . Formally, we have

$$h_t = f(W \cdot x_t + V \cdot h_{t-1} + b_h) \quad (3)$$

where  $W$  and  $V$  are weight matrices, respectively.  $b_h$  is a bias term and  $f$  is a non-linear activation function.

In theory, RNN can keep a memory of previous information. However, it was difficult to train RNNs to capture long-term dependencies because the gradients tend to either vanish or explode. Therefore, some sophisticated variants of RNN were proposed. Long-short term memory units are proposed in Hochreiter and Schmidhuber (1997). This model introduces a gating mechanism, which controls the proportions of information to forget and to pass on to the next time step. Concretely, the LSTM-based recurrent neural network comprises four components: an input gate  $i_t$ , a forget gate  $f_t$ , an output gate  $o_t$ , and a memory cell  $c_t$ . LSTM memory cell is implemented as following:

$$\begin{aligned}
f_t &= \sigma(W_f \cdot x_t + U_f \cdot h_{t-1} + b_f) \\
i_t &= \sigma(W_i \cdot x_t + U_i \cdot h_{t-1} + b_i) \\
\tilde{C}_t &= \tanh(W_C \cdot x_t + U_C \cdot h_{t-1} + b_C) \\
C_t &= f_t \otimes C_{t-1} + i_t \otimes \tilde{C}_t \\
o_t &= \sigma(W_o \cdot x_t + U_o \cdot h_{t-1} + b_o) \\
h_t &= o_t \otimes \tanh(C_t)
\end{aligned} \quad (4)$$

LSTM can solve the long-distance dependencies problem to some extent. However, the LSTM’s hidden state  $h_t$  takes information only from the past, knowing nothing about the future. An elegant solution whose effectiveness has been proven by previous work (Dyer et al., 2015) is bi-directional LSTM(BLSTM). The basic idea is to present each sequence forwards and backwards to two separate hidden states to capture past and future information, respectively. Then the two hidden states are concatenated to form the final output. In this paper, we employ a bi-directional LSTM, which could capture the contextual information of the current input, to predict the abbreviations of full terms. Since we give a specific segmentation tag and a pos tag for every character, each segmentation tag and pos tag can

be mapped to a real-valued vector by looking up their own embedding tables. These embeddings and character embeddings are all initialized randomly. At current time-step  $t$ , the character embedding, segmentation tag embedding and pos tag embedding are concatenated as the input  $x_t$ . Embeddings of segmentation tag and pos tag are both 20-dimensional. Character embedding is 50-dimensional. The hidden layer size of BLSTM is 200, 100 for forward LSTM and 100 for backward LSTM.

## 4. Evaluation

### 4.1. Evaluation Metrics

For evaluating abbreviation prediction quality, the systems are evaluated using the following two metrics:

- **Discriminate accuracy:** The discriminate accuracy checks the accuracy of discriminating positive and negative full forms, without comparing the generated abbreviations with the gold-standard abbreviations.
- **All-match accuracy (All-Acc):** The number of correct outputs (i.e., label strings) generated by the system divided by the total number of full forms in the test set.
- **Character accuracy (Char-Acc):** The number of correct labels (i.e., a classification on a character) generated by the system divided by the total number of characters in the test set.

### 4.2. Simple Heuristic Baseline System

The simple heuristic system means always choosing initial characters of words in the segmented full form. This is because the most natural abbreviating heuristic is to produce the first character of each word in the original full form. This is just the simplest baseline.

### 4.3. Evaluation

To study the performance of other machine learning models, we also implement other well known sequential labeling models, including maximum entropy Markov models (MEMMs) (McCallum et al., 2000) and averaged perceptrons (Perc) (Collins, 2002). Besides these traditional models, we also implement a bidirectional LSTM(BLSTM) to evaluate the performance of neural networks on this task.

The experimental results are shown in Table 3. In the table, the overall accuracy is most important and it means the final accuracy achieved by the systems in generalized abbreviation prediction with NFFs. For the completeness of experimental information, we also show the discriminate accuracy. The CRF model outperforms the MEMM

and averaged perceptron models. The CRF model achieves best overall character accuracy. BLSTM outperforms other models in both discriminate accuracy and all-match accuracy. However, training a neural network always needs a large amount of data. With a dataset that is not so large, the ability of a neural network may be limited.

## 5. Conclusions and Future Work

This paper proposes a novel abbreviation prediction dataset with NFFs. Different machine learning methods are evaluated on this general abbreviation task. LSTM shows competitive performance in this task. However, neural networks usually need large data for training. The related corpora are not sufficient and researches for general abbreviation prediction using neural networks are encouraged.

## 6. Acknowledgements

This work was supported in part by National Natural Science Foundation of China (No. 61673028), National High Technology Research and Development Program of China (863 Program, No. 2015AA015404), and an Okawa Research Grant (2016). Email correspondence to Xu Sun.

## 7. Bibliographical References

- Chang, J.-S. and Lai, Y.-T. (2004). A preliminary study on probabilistic models for chinese abbreviations. In *Proceedings of the Third SIGHAN workshop on Chinese language learning*, pages 9–16.
- Chang, J.-S. and Teng, W.-L. (2006). Mining atomic chinese abbreviations with a probabilistic single character recovery model. *Language Resources and Evaluation*, 40(3-4):367–374.
- Collins, M. (2002). Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, pages 1–8. Association for Computational Linguistics.
- Dyer, C., Ballesteros, M., Ling, W., Matthews, A., and Smith, N. A. (2015). Transition-based dependency parsing with stack long short-term memory. *arXiv preprint arXiv:1505.08075*.
- Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.
- Lafferty, J., McCallum, A., and Pereira, F. C. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data.
- McCallum, A., Freitag, D., and Pereira, F. C. (2000). Maximum entropy markov models for information extraction and segmentation. In *Icml*, volume 17, pages 591–598.
- Park, Y. and Byrd, R. J. (2001). Hybrid text mining for finding abbreviations and their definitions. In *Proceedings of the 2001 conference on empirical methods in natural language processing*, pages 126–133.
- Schwartz, A. S. and Hearst, M. A. (2002). A simple algorithm for identifying abbreviation definitions in biomedical text. In *Biocomputing 2003*, pages 451–462. World Scientific.
- Sun, X. and Wang, H. (2006). Chinese abbreviation identification using abbreviation-template features and context information. In *Computer Processing of Oriental Languages. Beyond the Orient: The Research Challenges Ahead, 21st International Conference, ICCPOL 2006, Singapore, December 17-19, 2006, Proceedings*, pages 245–255.
- Sun, X., Wang, H.-F., and Wang, B. (2008). Predicting chinese abbreviations from definitions: An empirical learning approach using support vector regression. *Journal of Computer Science and Technology*, 23(4):602–611.
- Sun, X., Okazaki, N., and Tsujii, J. (2009). Robust approach to abbreviating terms: A discriminative latent variable model with global information. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*, pages 905–913. Association for Computational Linguistics.
- Sun, X., Li, W., Meng, F., and Wang, H. (2013a). Generalized abbreviation prediction with negative full forms and its application on improving chinese web search. In *IJCNLP*, pages 641–647.
- Sun, X., Okazaki, N., Tsujii, J., and Wang, H. (2013b). Learning abbreviations from chinese and english terms by modeling non-local information. *ACM Trans. Asian Lang. Inf. Process.*, 12(2):5:1–5:17.
- Tsuruoka, Y., Ananiadou, S., and Tsujii, J. (2005). A machine learning approach to acronym generation. In *Proceedings of the ACL-ISMB Workshop on Linking Biological Literature, Ontologies and Databases: Mining Biological Semantics*, pages 25–31. Association for Computational Linguistics.
- Wren, J. D., Garner, H. R., et al. (2002). Heuristics for identification of acronym-definition patterns within text: towards an automated construction of comprehensive acronym-definition dictionaries. *Methods of information in medicine*, 41(5):426–434.
- Yang, D., Pan, Y.-c., and Furui, S. (2009). Automatic chinese abbreviation generation using conditional random field. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Short Papers*, pages 273–276. Association for Computational Linguistics.
- Yang, D., Pan, Y.-C., and Furui, S. (2012). Vocabulary expansion through automatic abbreviation generation for chinese voice search. *Computer Speech & Language*, 26(5):321–335.
- Zhang, L., Li, L., Wang, H., and Sun, X. (2014a). Predicting chinese abbreviations with minimum semantic unit and global constraints. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1405–1414.
- Zhang, L., Wang, H., and Sun, X. (2014b). Coarse-grained candidate generation and fine-grained re-ranking for chinese abbreviation prediction. In *Proceedings of the 2014*

*Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1881–1890.

Zhang, Q., Qian, J., Guo, Y., Zhou, Y., and Huang, X. (2016). Generating abbreviations for chinese named entities using recurrent neural network with dynamic dictionary. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 721–730.