Computational Modelling of Plurality and Definiteness in Chinese Noun Phrases

MSc Thesis

COMPUTING SCIENCE

DEPARTMENT OF INFORMATION AND COMPUTING SCIENCES

Author:
Yuqi Liu (5383986)

First Supervisor:
Dr. Guanyi Chen
Second Supervisor:
Prof. dr. C.J. (Kees) van Deemter

June 2023

Abstract

Theoretical linguists have suggested that some languages (e.g., Chinese and Japanese) are "cooler" than other languages based on the observation that the intended meaning of phrases in these languages depends more on their contexts [1]. As a result, many expressions in these languages are shortened, and their meaning is inferred from the context. In this paper, we focus on the omission of the plurality and definiteness markers in Chinese noun phrases (NPs) to investigate the predictability of their intended meaning given the contexts. To this end, we built a corpus of Chinese NPs, each of which is accompanied by its corresponding context, and by labels indicating its singularity/plurality and definiteness/indefiniteness. We carried out corpus assessments and analyses. The results suggest that Chinese speakers indeed drop plurality and definiteness markers very frequently. Building on the corpus, we train a bank of computational models using both classic machine learning models and state-of-the-art pre-trained language models to predict the plurality and definiteness of each NP. We report on the performance of these models and analyze their behaviours.

Contents

1 Introduction

Computational Linguistics (CL) is one of the major research areas of Artificial Intelligence. Much research in CL focuses on practical applications. People follow this line design programs that automatically process human language (natural language), such as string matching, translation systems, search engines, etc. Moreover, CL, as an interdisciplinary research area, also studies how to build computational models for linguistic phenomena or models of linguistic theories [2–4]. As a result, we are able to empirically validate hypotheses proposed by theoretical linguists, on the one hand. On the other hand, we have a better understanding of these phenomena and theories from a computational perspective. In this thesis, we focus on the computational model of one vital issue in Chinese linguistics: the comprehension of plurality and definiteness in Chinese Noun Phrases (NPs).

In what follows, we first elaborate on the motivation of why we were interested in this subject matter and its background. Then, we introduce the research questions of the current study and what we did in response. At length, we depict the structure of this thesis.

1.1 Motivation

In English, an English NP's plurality and definiteness are always conveyed through explicit markers. Precisely, the plurality of English NPs decided by using inflectional morphology. For example, English speakers inflect noun "*book*" using a plural marker to "*books*". Definiteness is often expressed using articles. For example, the definite article "*the*" in the phrase "*the book*" indicates that the identity of the "book" is known to the reader.

Nevertheless, different from English, in Chinese, plurality and definiteness are often expressed implicitly. In other words, a bare noun in Chinese can be either definite or indefinite and either singular or plural.

Consider the following example of the noun "狗 (gǒu)" (*dog*):

(1) a. 狗 很 聪明。
gǒu hěn cōngmíng
Dogs are intelligent.

b. 我 看到 狗。
wǒ kàndào gǒu
I saw a dog/dogs.

c. 狗 跑 走 了。
gǒu pǎo zǒu le
The dog(s) ran away.

The word "狗 (gǒu)" in the sentence (1-a) makes a general reference and, thus, is translated as "*dogs*". In the sentence (1-b), the "狗 (gǒu)" is an indefinite noun, but whether it refers to a single dog or a set of dogs needs to be decided by wider contexts. Likewise, the plurality of the "狗 (gǒu)" in the sentence (1-c) is also hard to be decided without further information, but, certainly, it is definite.

The above examples show us that, unlike English, plurality and definiteness are often omitted in Chinese. This is probably due to the hypothesis that Chinese is a "cool" language. In the next subsection, we describe what "cool" language is, and an example of plurality and definiteness are inferred by the context in Chinese.

1.2 Background

In theoretical linguistics, it has been noted that speakers have to weigh clarity and conciseness [5], and that different languages handle this trade-off differently [6]. Someone elaborated this idea by hypothesizing that some languages (especially, Eastern Asian languages, e.g., Chinese and Japanese) are "cooler" than other languages in that the intended meaning depends more on contexts [7]. As a consequence, speakers of these "cool" languages often (discursively and naturally) omit pronouns in pursuit of brevity and assume listeners can infer them from their contexts. Later on, the theory was extended, suggesting that many components in cool language are omittable, such as plurality markers, aspect markers, and definiteness markers, some of which have been investigated in the realm of computational linguistics [8–12].

To graphically illustrate how native Chinese speakers infer plurality and certainty from context, consider the following example:

(2) a. 小明 养 了 只 狗，很 聪明。
xiǎomíng yǎng le zhī gǒu, hěn cōngmíng
Xiaoming has a dog that is very smart.
b. 我 昨天 看到 小明 遛 狗 了。
wǒ zuótiān kàndào xiǎomíng liù gǒu le
I saw Ming walking the dog yesterday.

In this example, we can reckon that the sentence (2-a) is the context of the sentence (2-b). Based on the content of the sentence (2-a), it is easy to know that the "狗 (gǒu)" in the sentence (2-b) refers to Xiaoming's dog, rather than a general reference. Moreover, the quantifier "一 (yì)" is omitted in the sentence (2-a). When we add this quantifier to the original text, it is clear that the word "狗 (gǒu)" in the sentence (2-b) should correspond to a single dog. It is clear from these examples that Chinese native speakers can infer the plurality and the definiteness of Chinese NPs through the context in the Chinese corpus. This phenomenon of inferring plurality and definiteness from context is what we want to analyze by computational models. In the next subsection, we define exactly what our research questions are.

1.3 Research Questions

As mentioned above, in this thesis, we focus on the computational model of the comprehension of plurality and definiteness in Chinese NPs. So, we have the following research questions. **First**, from the comprehension perspective, we tried to empirically validate hypothesis proposed by theoretical linguists that plurality and definiteness are omitted in Chinese due to the "cool" hypothesis. We wanted to know *How frequently are plurality and definiteness omitted in Chinese?* **Second**, from the computational perspective, we expected to have a better understanding of this hypothesis. We planned to build computational models to understand *How well are neural models (which are thought to capture context quite well) able to predict information about plurality/definiteness in Chinese Noun Phrases?* In other words, we planned to construct computational models to predict, for a given noun phrase in a given discourse context, whether the noun phrase is singular or plural and whether it is definite or indefinite.

1.4 This Study

We divided the study into two steps, dataset construction, and computational modeling.

Dataset construction is to construct a Chinese dataset in which each NP is annotated with its plurality and definiteness[1]. Constructing such a dataset is a challenging task, but we can use a parallel corpus to accomplish it. Since the plurality and definiteness are not omitted in English, we can follow the NPs in English to correspond to these components in Chinese NPs. The general process of construction is as follows (please see chapter 3 for details): **First**, we conducted word alignment and NPs identification on the Chinese corpus and the English corpus; **Second**, we did NPs matching to paired Chinese NPs and English NPs, and post-processing to select the pairs that make sense for this study; **Third**, we annotated the Chinese NPs with its plurality and definiteness. After the dataset construction, we counted the percentage of samples in which plurality and definiteness were omitted in the dataset to check if the phenomenon of plurality and definiteness being omitted occurred in our dataset. We found that it is common for native Chinese speakers to omit plurality and definiteness markers in the Chinese corpus. Moreover, in order to have a better understanding of plurality and definiteness from the comprehension perspective, we also verified on the dataset some linguistic papers on plurality and definiteness of Chinese NPs. For example, Native Chinese speakers are always accustomed to using "们 (men)" to determine the plurality of NPs, but linguists have confirmed that "们 (men)" is not a plural marker, but a collective marker. We were very interested in this controversial view, and explored whether "们 (men)" is a plural maker in our dataset. To this end, We found that it is doubtful to see "们 (men)" as a plural maker.

In addition, after the construction of the dataset, to check if the dataset is of good quality to be used to build the computational models, we hired some native Chinese speakers to assess the dataset. We consider the quality of the dataset was good enough for computational models. During the

[1] The dataset construction completed in our preliminary work, but for the sake of completeness of the thesis content, we present this part of the work again. In this study, we specifically explored how much plurality and definiteness are omitted in this dataset and verified some interesting linguistic perspectives.

evaluation of the quality of the dataset, we also have some interesting findings. We tested different evaluation strategies and found that different evaluation strategies produced significantly different results for annotators evaluating the quality of dataset annotation definiteness due to the Framing Effects in human evaluation [13]. We also found that native Chinese speakers are very insensitive to the definiteness of NPs, which is consistent with the view of linguists.

After dataset construction, We built the computational models to predict the plurality and the definiteness. Specifically, we tackled predicting the plurality and the definiteness of Chinese NPs as two classification tasks, i.e., classifying a Chinese NP as plural or singular and as definite or indefinite. As mentioned earlier, Chinese is a “cool” language, and both the plurality and the definiteness of Chinese NPs may be contained in the context. Therefore, we selected some state-of-art classification algorithms for understanding context (e.g., Recurrent Neural Network, BERT, RoBERTa) and build the computational models. Moreover, we also compared the performance differences between these algorithms and some classic machine learning algorithms (e.g., Logistic Regression, Support Vector Machine, Random Forest) in terms of their ability to predicting plurality and definiteness. From the experiments, we found that state-of-art classification algorithms perform better than classic machine learning algorithms due to a better ability to capture the contexts, and the models perform worse for the prediction of the definiteness than for the plurality, which is consistent with the performance of the humans (for more details, please see Section 5). Further, we did three post-hoc analyses to further explore the predictive performance of computational models for plurality and definiteness. We had some interesting founding as follows: (1) we confirmed that as the length of the context increased, the worse the performance of the computational model; (2) we found that models can significantly benefit from predicting plurality and definiteness simultaneously compared to predicting them separately; (3) models predicted of both tasks on explicit expressions are easier than on implicit expressions.

1.5 The Structure of this Thesis

The rest of this thesis is structured as follows. In section 2, we describe the related work of this project, including the related research about classification algorithms, the details of classification algorithms we used in this study, and the related linguistics research of plurality and definiteness of Chinese NPs. In section 3, we describe the construction of the dataset in detail. Section 4 describes the computational models we used. Section 5 illustrates and compares their performance on our dataset. In section 6, we report our further analyses of these computational models. Finally, in section 7 we conclude all the work as well as the limitations and provide an outlook for future work.

2 Related Work

In this section, we present the related work to this project from three perspectives. We first introduce the development of classification algorithms that can capture context to give us a reference for algorithms selection. Then, we describe the details of our chosen classification algorithms. Finally, we present linguistic research related to the plurality and the definiteness of Chinese NPs, and illustrate linguistic perspectives on the phenomenon of plurality and definiteness of Chinese NPs expressed implicitly.

2.1 Classification Algorithms for Capturing Context

Research on context can be traced back as far as n-gram features [14]. Specifically, it treats n adjacent words as a feature and records whether the feature appears in the text. With this method of generating features, some machine learning algorithms such as Support Vector Machine (SVM) [15] and Logistic Regression (LR) [16] can have a very limited ability to capture context. But, n-gram features have significant limitations, it makes the features very sparse, which can significantly reduce the performance of the algorithm, and it does not incorporate the idea of capturing the context into the algorithm.

Recurrent Neural Network (RNN) is a neural network model that was proposed earlier to be able to understand the order of data, that is, have the ability to capture the context [17]. However, it has the problem of not being able to handle data that is too long in sequence. Subsequently, a series of improved tricks was proposed to solve this problem, such as LSTM and GRU [18, 19]. But, both LSTM and GRU still suffered from the problem of only being able to read sequences in one direction. Of course, in subsequent research on model design, many more models have been developed that can learn the sequence data from two directions, such as bidirectional LSTM (Bi-LSTM) [20]. In any case, the performance of models at that moment has not taken a qualitative leap forward in the ability to capture the context.

Attention has been proposed in recent years, along with an important model, the Transformer [21]. It solves the problem that the models cannot learn sequence data in both directions and solves the gradient vanishing problem of RNN. Moreover, it also tackles the problem that RNN cannot be computed in parallel, which greatly shortens the training time of the algorithm.

In addition to the breakthroughs in research on capturing context in recent years, the idea of solving multiple tasks with one model is becoming more and more mainstream. This is the idea of pre-trained models. Specifically, a pre-trained model is considered a meta-model for understanding the text. This model can be used for different NLP tasks (also called downstream tasks), and the tasks could be completed after supervised fine-tuning using the corresponding labeled data.

The study of word vectors is a very early study related to the idea of pre-training language models. The word vector at the beginning did not make changes to the vector representation of words depending on the context of different sentences, as in Word2Vec [22] and Glove [23]. ElMo was then proposed as a model that allows different vector representations of words according to different contexts [24]. It is often involved in the whole training process as part of the model. Its emergence led to the development of the idea of word vectors into the idea of pre-trained models.

Combining the idea of self-attention and the idea of pre-training, Bidirectional Encoder Representation from Transformers (BERT) emerges as a powerful pre-trained language model [25]. It has good context capture capability and can be used as a classification algorithm by fine-tuning.

Another pre-trained model with powerful context capture capability is the Generative Pre-training model (GPT), which evolved from the decoder part of Transformer [26]. As it evolves from GPT-1 to GPT-3, it may also be able to be used as a classification algorithm. Unlike other models, it uses text generation to accomplish the classification task. Specifically, people can design the corresponding prompts with appropriate examples. In this way, the model can perform few-shot learning in forward propagation to generate the result of classifies [27].

2.2 The Details of Classification Algorithms

As mentioned in Section 1, we used a variety of classification algorithms, including machine learning (ML), Bi-LSTM, and pre-trained language models (PLMs). In this subsection we provide a detailed introduce of these classification algorithms in this study.

2.2.1 Machine Learning

In this study, we selected three machine learning algorithms, Random Forest (RF) [28], Support Vector Machine (SVM) [15], and Logistic Regression (LR) [16]. Below we describe each of them individually.

Random Forest is an ensemble algorithm using the idea of bagging [28]. In simple terms, RF can be thought of as algorithms that vote on the results of multiple decision trees.

Each decision tree is a tree-structured model. It divides the dataset into subsets by finding the value of a feature, aiming for samples in each subset to contain only one label or the vast majority of samples with the same label. Classification of a new instance is done by moving the instance through the tree based on the feature values, starting at the root node and ending up at one of the leaf nodes. The instances are then classified into the majority class of this node. A simple example describing a decision tree is shown in Figure 2.1 (this decision tree is used to predict whether a customer wants a soft drink or a beer). In this example, each sample of data would contain three features, age, whether or not taking the cephalosporin, and whether or not wanting a beer. This tree splits the data into two classes, soft drinks and beer, based on these three features.

Although the decision tree can be interpreted well and efficiently, its performance is strongly influenced by the training set and its robustness is poor. When the training set is not of high quality, it can suffer from overfitting, resulting in very high variance. Aiming to solve this problem, RF based on the idea of bagging was proposed.

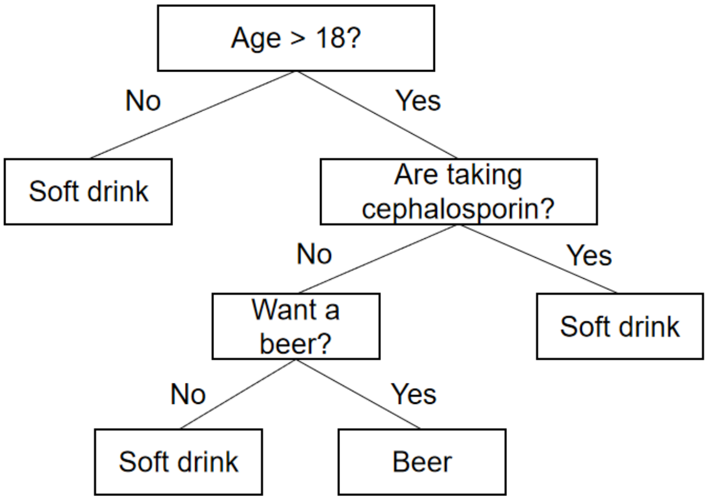


Figure 2.1 Example decision tree for classification (not based on actual results or data)

RF is the combination of multiple decision trees. For each decision tree in RF, its training data are obtained by randomly sampling the original data with replacements, this method is also called bagging. In addition, instead of using all the features in the original data when generating each decision tree, the RF also randomly draws some of the features with replacements. To classify an instance, RF is voted by all its decision trees and the label with the most votes is the prediction. By randomly sampling data as well as features and voting, the robustness of RF is improved and its variance is reduced, allowing it to often perform better than individual decision trees. Although RF is not as easy to interpret as a single decision tree, it still has good interpretability. This is an advantage for the NLU tasks, by interpreting the model one can better understand the behavior of the model. The most significant disadvantage of RF is that as the task becomes more and more complex, RF requires a large amount of training data as well as a large amount of computational resources.

Support Vector Machine refers to finding an optimal hyperplane in the feature space, partitioning the data according to the labels, and the minimum distance of data in the feature space belonging to different labels from this hyperplane should be as large as possible [15]. This algorithm is very intuitive in that data belonging to different classes should be as far away from the decision surface as possible.

It has been shown that it is always possible to separate two classes by hyperplanes using non-linear mappings in sufficiently high-dimensional feature spaces [29]. This means that hyperplanes can be found by mapping the data into higher dimensional spaces via non-linear transformations. This nonlinear mapping is known as the kernel function in SVM. A concrete example is shown in Figure 2.2 [2]. By mapping the data from 2D to 3D using the kernel function, a hyperplane (decision surface) can be found. For a new instance, the SVM decides which class it belongs to based on the

[2] Source of image: https://www.cnblogs.com/tabball/p/13175547.html

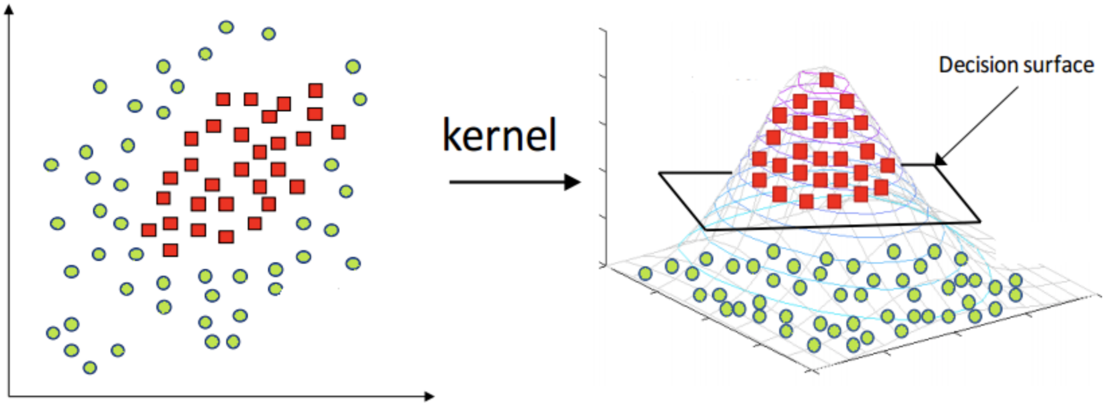


Figure 2.2 Example of finding a hyperplane in the higher dimensional space

hyperplane after mapping it to the corresponding feature space via the same kernel function.

Compared with other ML-based models, SVM is more suitable for handling high-dimensional data and can achieve good results even if the number of dimensions is larger than the number of training data. For NLP tasks, whether word embedding or N-gram, the number of features is very large, so SVM is very suitable. A significant drawback of SVM in this study is that it is not interpretable.

Logistic Regression is based on linear regression to solve the classification problem. It completes the classification task by transforming the values of linear regression to labels through the logistic response function [16].

There has a issue with linear regression being used directly for classification: linear function is not guaranteed to produce values between 0 and 1. LR eschews direct regression on probabilities and instead maps the values of linear regression to between 0 and 1 via a logistic response function, thus solving this problem. The logistic regression function is shown in Equation 1, where t represents the true label, $\mathbf{x}$ represents the values of features, and $\mathbf{w}$ represents the weights of features in linear regression. With the logistic response function, LR maps the results of the linear regression between 0 and 1. One might ask why this form of the regression response function is used instead of a simpler one if it is just mapping the linear regression results between 0 and 1. The logical response function shown in Equation 1 is also known as the sigmoid function, the two-label classification task is consistent with the Bernoulli distribution, and this logical response function is derived based on the Bernoulli distribution.

$$\mathbb{E}(t) = (1 + e^{-\mathbf{w}^T\mathbf{x}})^{-1} \tag{1}$$

For a new instance, LR computes $\mathbb{E}(t)$ of this instance and compares it to 0.5 to derive its predicted label. LR requires fewer computational resources compared to RF, and it also has well interpretability for the selection of features compared to SVM. The most significant downside of LR is that it would perform poorly due to the high-dimensional space and sparse data.

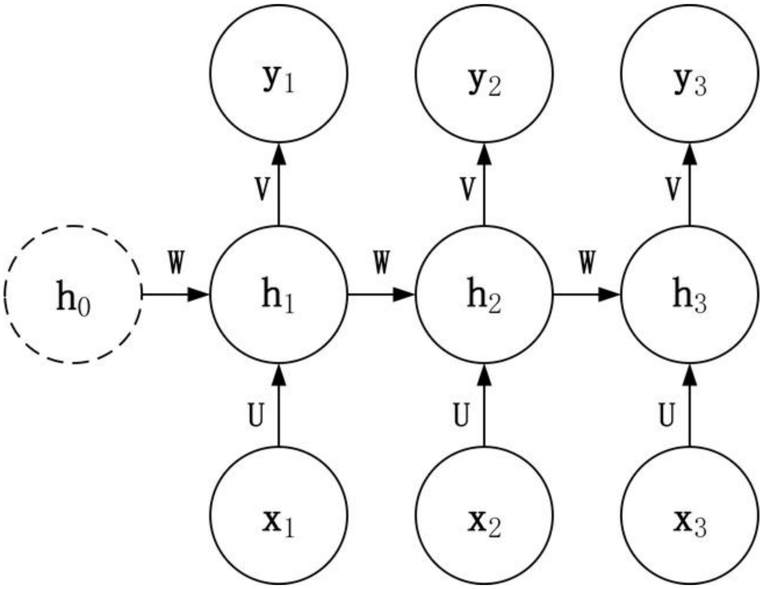


Figure 2.3 Architecture of simple RNN

2.2.2 Bi-LSTM

As mentioned above, RNN is the network that can learn the order of data. The architecture of RNN is shown in Figure 2.3. According to the figure, for each time-step t, RNN generates hidden vector h_t based on h_{t-1} and input x_t. The formula as shown in Equation 2, where σ represents the activation function (in this case the sigmoid function), W represents the weights of hidden vectors for generating the next hidden vector, and U represents the weights of inputs. The generation of the output y_t at time-step t is shown in Equation 3, where V represents the weights of hidden vectors for outputs. It is obtained by passing h_t through the softmax function.

$$h_t = \sigma(Wh_{t-1} + Ux_t) \tag{2}$$

$$y_t = Softmax(Vh_t) \tag{3}$$

With Figure 2.3, we can visualize that, for y_3, the gradient computation for x_1 goes through all time-steps when backpropagation through time. Due to the long derivation chain, the gradient of x_1 might be extremely small. This would result in a negligible effect of x_1 on y_3, and the RNN can only capture short-term dependencies (which is known as vanishing gradients). LSTM came into existence to solve the problem of vanishing gradients [18].

As shown in Figure 2.4 [3], an LSTM block has 4 main components: input gate, output gate, forget gate and memory cell. Through the collaboration of these four parts, the LSTM could selectively forget and retain information, and has reached the goal of storing long-term information. In par-

[3] Source of image: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7508408

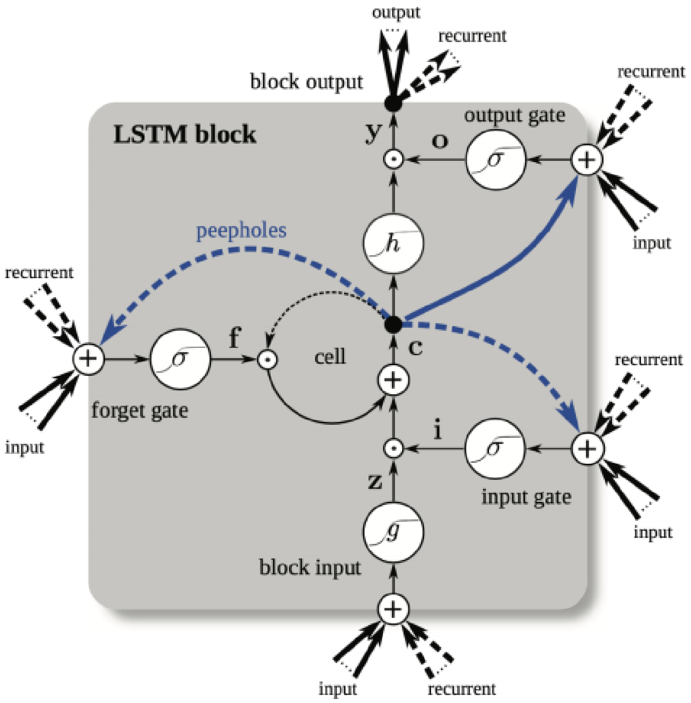


Figure 2.4 Architecture of an LSTM block

ticular, **input gate** is used to decide whether the input is passed on to the memory cell or ignored; **output gate** is used to decide whether the current activation vector of the memory cell is passed on to the output layer or not; **forget gate** is used to decide whether the activation vector of the memory cell is reset to zero or maintained; and **memory cell** is used to store the current activation vector.

As the name suggests, Bi-LSTM is a bi-directional RNN using LSTM, which consists of two LSTM layers from left to right and right to left for each hidden layer.

2.2.3 Pre-trained Language Models

PLMs are neural network models obtained by pre-training on massive textual data using unsupervised learning or self-supervised learning. They tend to learn the high-quality hidden representations of words and use them for a variety of specific NLP tasks (also called downstream tasks). In this study we mainly used the PLMs based on BERT, thus we describe BERT in detail below.

BERT is a pre-trained language model formed by overlaying the multiple encoders of Transformers. It obtains high-quality hidden representations of tokens through multi-task learning in the pre-training phase, and handles various downstream tasks by fine-tuning the model. The input sequence to BERT can be a sentence or a sentence pair. It splits an input text into tokens, with a special classification token ([CLS]) at the start. Aiming to differentiate the sentences in an input sequence, BERT uses two ways to separate them: (1) with a special token ([SEP]), and (2) by adding

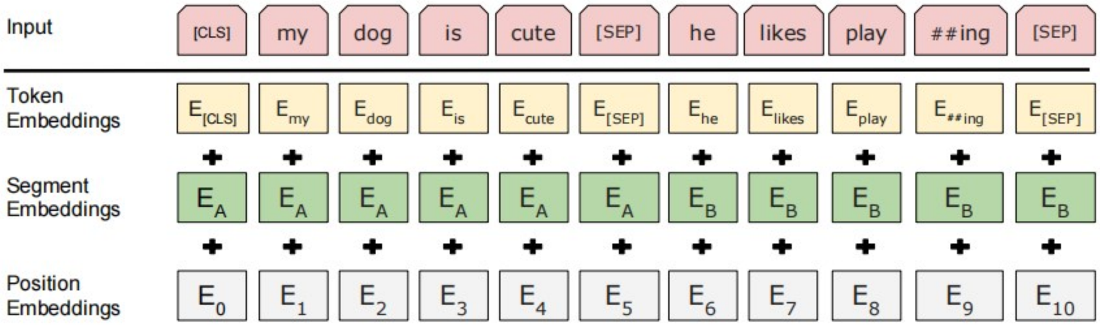


Figure 2.5 BERT input representation

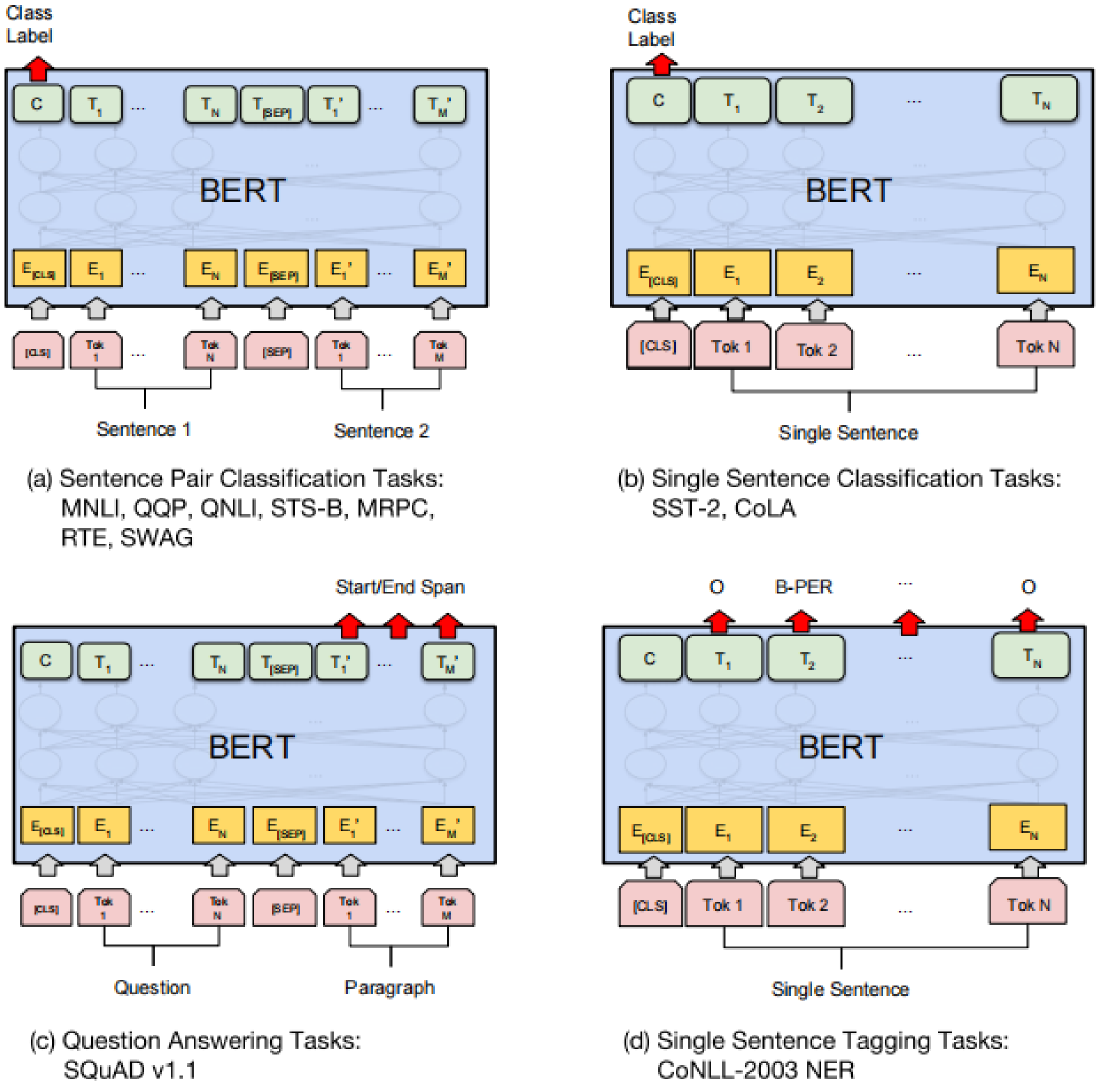
the segment embedding to indicate a token belongs to the first sentence or the second. For a given sequence, the architecture of tokens as shown in Figure 2.5 [4]. Each input token consists of three components: token embedding, segment embedding, and position embedding. **Token embedding** is obtained via WordPieces embeddings [30]. **Segment embedding** is used to separate sentences. **Position embedding** is used to store the location information for tokens. The input representation of BERT is constructed by summing the corresponding token, segment, and position embeddings.

In the pre-training phase, BERT uses multi-task learning to understand natural language. It uses two pre-training tasks: Masked LM (MLM) and Next Sentence Prediction (NSP). For **MLM**, BERT simply masks 15% of the input tokens at random, and then predict those masked tokens. During the fine-tuning phase, “[MASK]” does not appear in the input sequence, so to reduce the effect of “[MASK]” itself on the performance of BERT, there is a trick be adopted: If a token is chosen to be masked, BERT replaces this token with (1) the [MASK] token 80% of the time (2) a random token 10% of the time (3) the unchanged token 10% of the time. For **NSP**, it is to enable BERT to understand sentence relationships. Specifically, for two sentences, BERT needs to determine whether the second sentence is the next sentence of the first sentence. With these two pre-training tasks, BERT learns higher-quality hidden representations of tokens. At this point, BERT can be fine-tuned to perform a variety of downstream tasks.

In short, during the fine-tuning phase, for each downstream task, only the corresponding inputs and outputs need to be plugged into the BERT and fine-tuned all parameters. Examples of fine-tuning are shown in Figure 2.6 [5]. For Sentence Pair Classification Tasks, the task is accomplished by simply feeding the hidden representation corresponding to the [CLS] token into a linear layer to obtain the labels, and the same is true for Single Sentence Classification Tasks. For Question Answering Tasks, the task is accomplished by feeding the questions and paragraphs into BERT as sentence pairs, and then taking the second sentence of the output as the result. For Single Sentence Tagging Task, it is sufficient to use the hidden representation of each token as input to the linear layer to obtain tagging labels.

[4] Source of image: https://arxiv.org/abs/1810.04805

[5] Source of image: https://arxiv.org/abs/1810.04805



(a) Sentence Pair Classification Tasks: MNLI, QQP, QNLI, STS-B, MRPC, RTE, SWAG

(b) Single Sentence Classification Tasks: SST-2, CoLA

(c) Question Answering Tasks: SQuAD v1.1

(d) Single Sentence Tagging Tasks: CoNLL-2003 NER

Figure 2.6 Illustrations of fine-tuning BERT on different tasks.

2.3 Related Linguistic Research

In the following, we present the views of linguists on the definiteness and the plurality of Chinese NPs respectively, and list some interesting studies.

2.3.1 Definiteness

In English, the definiteness of an NP is a very common concept that is part of the semantics of NPs. The definiteness of the English NP is also expressed explicitly, often determined by the articles or some demonstratives (such as “this” and “his”) [31]. But whether Chinese NPs have the concept of definiteness, and how the definiteness of Chinese NP is expressed are questions that have been of interest to linguists. Hu first proposed that there is also a concept of definiteness in Chinese NPs [32]. Specifically, he inferred that there is definiteness in Chinese by finding that native speakers of Chinese have a habit that placing Chinese NPs in sentence-initial positions to indicate definiteness, as shown in (3-a). Although the concept of definiteness is present in Chinese NPs, many studies have shown that native Chinese speakers are not sensitive to definiteness [33]. This may be due to the fact that definiteness is often expressed implicitly in Chinese. Further, many linguists believe that the definiteness of Chinese NPs is often implicitly expressed due to Chinese being a kind of “cool” language. Yang laterally confirmed this idea. His experimental analysis of the types of errors Chinese people make in using English articles revealed that the most common error Chinese people make is the omission of the article [34].

(3) a. 水 开了
shuǐ kāile
water boiled

In addition to exploring whether the definiteness of Chinese NPs is expressed implicitly, linguists have also made some interesting points about the definiteness of Chinese NPs. For example, Bremmers et al. argue that in Chinese, bare nouns are often used to denote the definite NP when the time in the context is continuous [35]. This perspective suggests to us the possible conditions that native Chinese speakers implicitly express the definiteness of Chinese NPs. Another interesting research by Tarone et al. showed that different forms of the task had a significant effect on the accuracy of the judgment of the use of articles in English for Japanese speakers [36]. This perspective suggests that the extent to which the definiteness of NPs is embedded in context varies across different forms of the corpus.

These studies on the definiteness of Chinese NPs are very interesting. However, we did not make analyses of all these ideas in the dataset we constructed in this research. We include these analyses as part of future work.

2.3.2 Plurality

In Chinese, as in English, NPs have the concept of plurality. There are two forms of expressions for the plurality of Chinese NP: (1) at the lexical level, using numerals. (2) at the grammatical level, using plural grammatical markers such as “们 (men)” (equivalent to the ’s’ at the end of an English noun) [37]. However, unlike Indo-European languages, Chinese tend to express as briefly as possible. In other words, in Chinese, a number of plural NPs do not add grammatical markers [38]. Interestingly, Chinese speakers did not let the omission of the plural grammatical markers affect their judgment of the plurality of Chinese noun phrases. This may be because the plurality of Chinese noun phrases is already embedded in the context, and Chinese speakers can infer the plurality of noun phrases from the context.

In addition, linguists do not agree on the role of grammatical markers for the plural of Chinese NPs. For example, some people believe that “们 (men)” indicates plural in Chinese NPs, but others believe that “们 (men)” indicates a whole group (similar to the plurality case of “population” in English words) rather than plural [38, 39].

The analysis of “们 (men)” based on the constructed dataset is one of our works in this research. We will explain it in detail in Section 3.

3 Dataset

In our preliminary work, we solved one of the major challenges in this research: constructing a large-scale Chinese dataset in which each NP is annotated with its plurality and definiteness. It is very difficult to build such a dataset, the main reasons are (1) manually annotating large-scale data is very expensive, and (2) as mentioned above, many linguistic studies have shown that deciding the plurality and the definiteness of Chinese NPs (especially the definiteness) can be a challenging task even for native speakers [33].

Inspired by the study on pro-drop in machine translation systems Wang et al. [40], we can use the Chinese and English corpus to automatically annotate the plurality and the definiteness of Chinese NPs since English speakers will always convey these of English NPs explicitly. Such information can be found in various Chinese-English parallel corpora, especially those of relatively daily dialogues. More specifically, we took a parallel dialogues corpus as raw data and completed the construction of the dataset by word alignment, NP identification, NP matching, post-processing, and annotation. In the following, we detail the automatic dataset construction process and describe the resulting dataset and how we evaluate its quality.

3.1 Dataset Construction

Since we study the phenomenon of implicitly expressing Chinese NPs of plurality and definiteness, it is particularly important for the corpus to be able to realistically reflect the omission of plurality and definiteness by native Chinese speakers. News and other relatively formal genres partially sacrifice conciseness in the pursuit of clarity and are not suitable for this study. Therefore, we used a Chinese-English parallel corpus by Wang et al. [40]. They extracted more than 4.39 million sentence pairs from the subtitles of television episodes. In other words, the corpus consists of dialogues, the Chinese part of which is more in line with the way native Chinese speakers use Chinese. In this study, we used a total of 2.15 million sentence pairs to construct the dataset.

The dataset construction can be divided into five phases: (1) align Chinese and English words; (2) recognize the NPs for each of the Chinese and English sentences; (3) match Chinese NPs and English NPs; (4) filter out the NPs we are not interested in; (5) annotate Chinese NPs according to English NPs.

The above phases result in a dataset of Chinese NPs annotated with definiteness and plurality. There are approximately 1.04 million data in the dataset. Below we present the details of each phases.

Word alignment. We used GIZA++[6] [41] to propose word alignments for each pair of sentences in the parallel corpus. Since automatic word alignment is asymmetric (i.e., alignments from English tokens to Chinese tokens are sometimes different from alignments from Chinese tokens to English tokens), we recorded the alignment proposals by GIZA++ in both “directions”. The word alignment

[6] http://fjoch.com/GIZA++.html

module only aligns two tokens if they are aligned by GIZA++ in both directions. While this would substantially reduce the number of alignment tokens, increasing confidence in the alignment results will prevent a lot of noise generated during the word alignment process.

NP recognition In this phase, we used CoreNLP[7] [42] to identify NPs in both English and Chinese. Specifically, we obtained the syntactic tree for each sentence in Chinese and English using CoreNLP. Then, we extracted all the NPs from the syntax tree. We also recorded the part-of-speech (POS) tag for each word in this step for subsequent use.

NP matching There is a simple intuition about NPs matching algorithm: an NP in the source language is paired with the NP in the target language that has the most aligned tokens. We designed a simple but effective method in which there are two steps: **First**, for each direction, an NP in the source language is paired with the NP in the target language that has the most aligned words with it. But in this case, it is still possible that an NP in the source language would match multiple NPs in the target language with the same number of aligned tokens. For these NPs in the target language, we distinguished which NPs are constituents of other NPs and which are not based on syntactic. NPs were not the constituents of other NPs were counted as paired. On the other hand, for an NP with other constituents NPs, only the shortest NP was counted as paired. **Second**, a match was done if and only if a pair of NPs were paired in both directions.

Post-processing After the NPs matching, since NPs tend to have a large number of nested structures and them are not of interest to us, we performed three more post-processes on the matched results: removing conjunctions,ignoring the inner structure, and filtering pronouns: **First**, we removed all NP conjunctions and only kept their constituents. For example, the NP “Tom and Jerry” contains two NPs. We remove it and keep only “Tom” and “Jerry”. This we were interested in plurality and definiteness for each of juxtaposed NP rather than NP conjunctions; **Second**, for each NP, we dropped all its constituents. For example, if all of the “Jerry’s Cheese”, “Jerry” and “Cheese” are matched in the previous step, we only keep “Jerry’s Cheese” in our dataset. This is because we were not interested in the plurality and the definiteness of these constituents; **Third**, we also removed all pronouns, as they were not the focus of this study.

To show NPs matching and Post-processing more clearly, the pseudo code is shown below. $Count$ is used to count the number of aligned tokens for a pair of NPs. $FindMatchedNP$ is used to obtain one-way matching results. $TMatch$ is used to obtain two-way matching results. $ConjDel$ is used to remove all NP conjunctions and only keep their constituents. $InnerDel$ is used to remove the constituents for each NP (without NP conjunctions). $PronDel$ is used to remove the pronouns.

Annotation In terms of annotation, we relied on some simple grammar rules and keywords to obtain the plurality and definiteness of English NPs and then annotated the plurality and definiteness to the corresponding Chinese NPs. Concretely, we determined what part of each English NP was the head noun by some simple grammar rules. For example, often the part of an English NP before the preposition is more likely to be the head noun. Subsequently, we determined the plurality and the definiteness of each English NP. We annotated an English NP as plural if: (1) it has a plural POS tag (i.e., NNS or NNPS); (2) it is a numeral phrase that specifies a quantity larger than one (i.e., “two cups of coffee”). Otherwise, it is a singular NP. For the definiteness of an NP, the annotation

[7] https://stanfordnlp.github.io/CoreNLP/

Algorithm 1 NP Matching and Post-processing

```
Input: word alignment: $WA$, English NPs: $ENPs$, Chinese NPs: $CNPs$
Output: NPs matching results: $NP\_Match$
  for $NP$ in $ENPs$ do
      $EC\_Match \Leftarrow Count(NP, CNPs, WA)$
      $CNP\_M \Leftarrow FindMatchedNP(EC\_Match)$
      $SECMatched \Leftarrow SECMatched \cup \{(NP, CNP\_M)\}$
  end for
  for $NP$ in $CNPs$ do
      $CE\_Match \Leftarrow Count(NP, ENPs, WA)$
      $ENP_M \Leftarrow FindMatchedNP(CE\_Match)$
      $SCEMatched \Leftarrow SCEMatched \cup \{(NP, ENP_M)\}$
  end for
  $NP\_Match \Leftarrow TMatch(SECMatched, SCEMatched)$
  $NP\_Match \Leftarrow ConjDel(NP\_Match)$
  $NP\_Match \Leftarrow InnerDel(NP\_Match)$
  $NP\_Match \Leftarrow PronDel(NP\_Match)$
  return $NP\_Match$
```

Table 1 The basic statistics of our dataset.

		PLURALITY		DEFINITENESS	
	Size	Singular	Plural	Definite	Indefinite
train	103686	79158	24528	48471	55215
dev	10368	7894	2474	4777	5591
test	10369	7925	2444	4844	5525

was done based on (1) its article; (2) whether it is a demonstrative phrase (i.e., whether it contains a demonstrative (decided based on its POS tag and its surface form), such as "*this*" or "*that*"); and (3) whether it is a proper name (i.e., an NP is definite if it is a proper name).

3.2 The Corpus

We annotated a total of 1.04 million data from the raw corpus to construct the dataset. Due to the limitation of computing resources, we randomly selected 10% of the data from the dataset for computational modelling (described in detail later in Section 4). Table 1 shows the basic statistics of the resulting dataset. We can see that a total of 124K annotated NPs were selected for computational model building. Three-quarters of this 124K data is marked as singular. Relatively the same amounts of data are labeled as definiteness and indefinite. There are 58K data marked as definiteness. Meanwhile, there are 66K data marked as indefinite. The statistics of the overall dataset are roughly consistent with this 124K data (the ratio of singular samples to plural samples is 7.5:2.5 and

Table 2 The label distribution statistics of “们 (men)” in the dataset.

	Definite	Indefinite	Total
Plural	2757 (32.12%)	5044 (58.76%)	7801 (90.88%)
Singular	347 (4.04%)	436 (5.08%)	783 (9.12%)
Total	3104 (36.16%)	5480 (63.84%)	

the ratio of definite samples to indefinite samples is 4.5:5.5 in the whole dataset). We then divided these 124K data into training, development and test sets in the ratio of 8:1:1.

Before building the computational models, we explored two questions related to this research based on the dataset: (1) As mentioned in section 2, since linguists believe that “们 (men)” is not a plural marker, we wanted to explore this perspective in the dataset; (2) We also wanted to know exactly how frequently do Chinese speakers express plurality or definiteness explicitly in this dataset.

3.2.1 Is “们 (men)” a plural marker?

Like the English plural markers (e.g. “+s”), the inflectional morpheme “们 (men)” has long been considered a plural marker in Chinese. But linguists have always agreed that “们 (men)” should be seen as a collective marker [38, 39], referring to a group of people or a collective (translated as “group of” or “set of”), because “们 (men)” is incompatible with number phrase. On the other hand, some linguists believe that Chinese NPs must considered as definiteness when they have the marker “们 (men)” [7]. These discussions about marker “们 (men)” were of great interest to us, so we explored these perspectives in our dataset.

We extracted all NPs whose head noun has a “们 (men)” suffix[8] and did statistics on the label distribution. The result showed on Table 2. For plurality, we found that while most of the NPs with the suffix ‘们 (men)” were marked as plural (approximately， 90.88%), a certain number of NPs were still marked as singular (approximately, 9.12%). This suggests that “们 (men)” can not be used as a plural marker. In terms of definiteness, the difference between the number of NPs with a “们 (men)” suffix marked as definite and indefinite was smaller (approximately 36.16% and 63.84% respectively), which contradicted the view of linguists that “们 (men)” can be a definiteness marker. For example, the “大人们 (dà rén men)” (*adults*) in the example (4-a) apparently has an indefinite reading. This embodied the conclusion that says “们 (men)” must be interpreted as definite is questioned.

(4) a. 大人们 会 告诉 你 并不是 这样。

[8] In other words, we remove NPs in which “们 (men)” does not function as suffixes, e.g., “哥们 (gē men)” (*brother*).

Table 3 The distribution statistics of express explicitly or implicitly in the dataset.

	Size	Explict	Implict
Plurality	1041676	129348 (12.42%)	912328 (87.58%)
Definiteness		165179 (15.86%)	876497 (84.14%)

dàrénmen huì gàosù nǐ bìngbúshì zhèyàng
Adults will tell you this is not the case.

3.2.2 How frequently do Chinese speakers express plurality or definiteness explicitly?

For each NP in the dataset, we annotated whether it expresses plurality or definiteness explicitly based on the POS tags and the parsing tree of the sentence in which this NP was located. For plurality, if the NP contains a numeral or a measure word, we marked it as express plurality explicitly. In terms of definiteness, we marked an NP express definiteness explicitly based on (1) it contains a proper name; (2) it contains a possessive; (3) there is a numeral or measure present, with a preceding demonstrative. It is worth noting that it is very difficult and impractical to design the best statistical rule to determine whether the plurality and the definiteness of NPs express explicitly. The above rule relaxes the restriction on expressing explicitly and goes for its upper limit.

The statistical results are shown in Table 3. From the result, we identified that merely 12.42% utterances convey plurality explicitly and 15.86% utterances contain explicit definiteness markers. This confirms that Chinese, as a "cool" language, its speakers indeed do not use plurality and definiteness markers very often.

3.3 Quality Assessment

After dataset construction, it is essential to assess the quality of the dataset to ensure that it can be used for the computational models. We manually assess its quality from the aspects of plurality and definiteness annotation as well as NP identification. We used two approaches to assess the dataset to obtain more reliable assessment results. Below we describe in detail the two assessment methods and analyze their results.

3.3.1 Assessment 1

We randomly sampled 400 samples from the dataset. After that, We hired 4 native Chinese speakers to assess the samples and ensured that each sample was judged by at least 2 annotators. They were asked to answer three questions (translate from Chinese): (1) Does the highlighted noun phrase correctly identified? (2) Is this a singular/plural (decided by the annotation in our corpus) phrase? and (3) Is this a definite/indefinite phrase?

After the experiment, we computed the accuracy and inter-annotator agreements (IAA). We computed two types of accuracy based on the number of annotators in agreement with the annotation. $\mathrm{Acc}_{=2}$ represents the percentage of samples where annotations by two annotators are consistent with the dataset, while $\mathrm{Acc}_{\geq 1}$ represents the percentage of samples where annotations by at least one annotator are consistent with the dataset. In terms of IAA, IAA(%) represents the percentage agreement and IAA(κ) represents the Cohen's Kappa [43]. It is worth noting that we did not compute Cohen's Kappa based on the raw human decisions, because for each question, the marginal distribution of one result is much larger than the other (e.g., "yes" is much more than "no"), which will cause the Cohen's Kappa to lose effect [44–46]. Instead, we translated their decisions in accordance with our labels. For example, if the label in our dataset is "plural" and the annotator provided a positive response, we assumed that the annotator annotated this sample as "plural"; On the other hand, if the annotator provided a negative response to a sample which is annotated by "plural" in our dataset, we assumed that he/she would annotate this sample as "singular". Then we got the label according to the translation to calculate Cohen's Kappa.

The assessment results are shown in Table 4 (in Appendix A we provide more detailed assessment results). From the results, the annotators generally agreed that the quality of NP recognition in this dataset was relatively high. $Acc_{\geq 1}$ was 96.25%, while $Acc_{=2}$ was 79.50%. One can ask why NP identification has received lower scores than the other two tasks. One major reason is that most identified incorrect NP identifications are about unsuccessfully including all modifiers (e.g., marking only "*the men*" from the true NP "*the man who is old*").

Regarding the plurality annotation task and the definiteness annotation task, the results show that the dataset has a high quality, with both $\mathrm{Acc}_{=2}$ being above 80%. It is evident from the IAAs that there are substantial agreements for all of three tasks by the annotators (in the three tasks, the value of IAA (%) is approximately 85% and the value of IAA (κ) is approximately 0.65). But there were some unreasonable results in the definiteness annotation task. The IAAs of this task are abnormally high. As mentioned before, Chinese native speakers are insensitive to the definiteness in Chinese NPs [33], which is inconsistent with the high IAAs results in this task. We guessed this is due to native Chinese speakers do not have a clear definition of definiteness, so they would simply agree with what they see, which is also known as the Framing Effects in human evaluation [13]. In other words, asking the annotator to answer the question using yes or no may influence the annotator's decision and create bias. This bias is further increased by the insensitivity of native Chinese speakers to the definiteness of Chinese NPs. Therefore, we conducted assessment 2 as a complement.

3.3.2 Assessment 2

To reduce the bias introduced by the framing effects, in assessment 2, we removed the labels from the dataset and kept only the highlighted NP identification information. We asked annotators to directly annotate the plurality and definiteness of the samples. Then we compared the annotator annotations with the labels in the dataset to obtain the assessment results. This time, we randomly

Table 4 Human Assessment Results, in which $Acc_{=2}$ represents the percentage of samples where annotations by two annotators are consistent with the dataset, $Acc_{\geq 1}$ represents the percentage of samples where annotations by at least one annotator are consistent with the dataset, IAA (%) is the percentage agreement and IAA (κ) is the Cohen's Kappa.

	Assessment 1				Assessment 2			
	$Acc_{=}2$	$Acc_{\geq}1$	IAA (%)	IAA (κ)	$Acc_{=}2$	$Acc_{\geq}1$	IAA (%)	IAA (κ)
NP Identification	79.50	96.25	0.8325	-	-	-	-	-
Plurality	84.00	96.75	0.8725	0.6477	74.00	85.50	0.8850	0.6679
Definiteness	81.00	97.25	0.8375	0.6731	53.00	77.50	0.7550	0.4755

selected 200 samples from the dataset and still guaranteed each sample was judged by at least 2 annotators. The results are still shown in Table 4.

Compared with the assessment 1, the $Acc_{=2}$ and $Acc_{\geq 1}$ of plurality and definiteness in this assessment declined. For the plurality, the $Acc_{=2}$ decreased 10 points, and the $Acc_{\geq 1}$ decreased 11.25 points. For the definiteness, the $Acc_{=2}$ and $Acc_{\geq 1}$ fell more significantly, with $Acc_{=2}$ downed by 28 points and $Acc_{\geq 1}$ downed by 20 points. However, the $Acc_{\geq 1}$ of plurality and definiteness remain around 80%, which indicates that the quality of the dataset can be guaranteed. As for consistency, there was no significant change in the IAAs of plurality, but the IAAs of definiteness decreased from significant, IAA (%) from 83.75% to 75.50%, and IAA (κ) from 0.67 to 0.47. This confirms the view of linguists that it is difficult for native Chinese speakers to recognize the definiteness of Chinese NPs.

3.3.3 Summary

Overall, we found some disagreements in the consistency of the annotators' assessment of the dataset. This may be due to the fact that it is more difficult for native Chinese speakers to recognize the definiteness of Chinese NPs, and the way how the questions are framed further amplifies this in the assessment results. Regardless, we believe that the dataset is still of high enough quality to be used for the computational models, since in the worst case in both assessments, about 80% of the samples are agreed upon by at least one person.

3.4 Limitations

Based on the construction process and assessment results of the dataset, we reckon that the dataset has the following limitations.

(1) As shown in the assessment experiments, disagreements exist in human annotation. This is true for many pragmatic tasks [47]. However, our automatic annotation strategy cannot take such agreements into consideration.

(2) Chinese NPs do not distinguish between countable and uncountable. In assessment 2 we found that the annotators would mark many NPs that are uncountable in English as plural. But these uncountable nouns are automatically marked as singular in the dataset. We would try to use a more reasonable way to deal with these uncountable nouns in future work.

(3) Both our automatic annotation and human assessments are precision-oriented. For example, we dropped the Chinese NP that did not match with any English NPs and, during the assessments, we only used NPs that had been matched. This makes our dataset overlook some Chinese NPs (this means that perhaps some of the NPs in the raw corpus that could be used in this study have been omitted by us) and we also did not calculate exactly how many such NPs we omitted from the assessment.

(4) In the assessments, we did not evaluate how the decisions of annotators would be influenced by providing them with additional contexts for each sample. This limitation was recognized because, as mentioned above, the meaning of a Chinese NP relies more on its context compared to its English counterpart.

4 Models

This section introduces the models we used to build our computational model. We used a variety of models that can capture context, from Bi-LSTM to state-of-art PLM-based models. To explore the gap between these models and classical machine learning algorithms in terms of their ability to predict the plurality and the definiteness, we also used three ML-based models to construct computational models as baselines. As follow, we would describe these models in detail.

4.1 ML-based Models

We have tried many classical ML-based models as the baseline. As stated in section 2, for ML-based models, designing the appropriate N-gram features gives them the ability to capture context. Specifically, we marked the target NP in each sample with the special character '*', as shown in (5-a), where "我的母亲 (wǒ de mǔ qin)" (*my mom*) is the target NP.

(5) a. 我 爱 * 我 的 母亲 *。
wǒ aì wǒ de mǔqin
I love my mom.

We did the word segment for each sample and then used N-gram ($N = 1, 2, 3, 4, 5$) to construct features for classification. As mentioned earlier, we used RF, SVM, and LR to build the computational models.

4.2 Bi-LSTM

As mentioned in section 2, RNN is considered the model dedicated to capturing context before PLMs. RNN suffers from the gradient disappearance/explosion problem, LSTM has been proposed to solve this problem. Based on that bi-directional LSTM (Bi-LSTM) that can capture the context in both directions has also been proposed[9] [20].

Previous studies showed that PLMs capture the context better than Bi-LSTMs due to the attention mechanism [21]. In this study, we reckoned that the differences in the ability of Bi-LSTM and PLMs to capture context had an impact on the performance of plurality and definiteness prediction.

The architecture of the Bi-LSTM we used is shown in Figure 4.1. Specifically, we used the Glove [23] to embed the Chinese characters as vectors for input. Then we obtained the intermediate representations of the tokens through the Bi-LSTM layer. After that, we contacted the intermediate representations of the first token and the last token of the target NP together and put them into a linear layer to obtain the prediction labels.

[9] The bi-directional capture context of Bi-LSTM can be understood as the combination of two unidirectional LSTM layers.

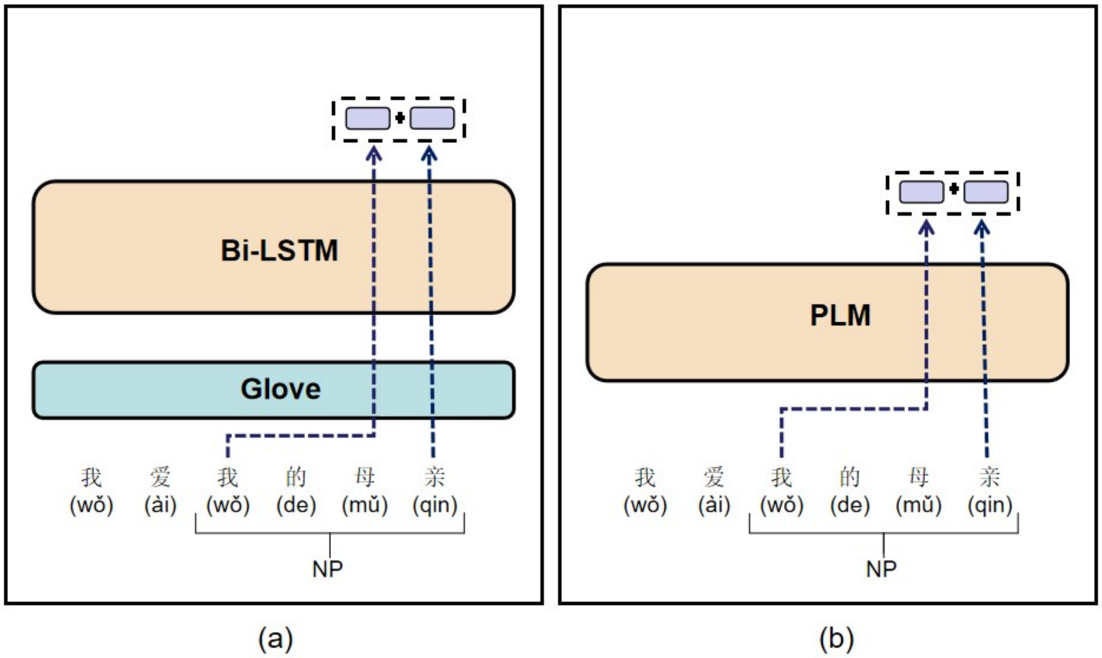


Figure 4.1 Illustration of the Bi-LSTM and PLMs Models. (a) the architecture of Bi-LSTM. (b) the architecture of PLMs.

4.3 PLM-based Models

Already mentioned above in section 2, the PLM model not only proposes more reasonable embeddings (better understanding of the text) but also improves the ability of the model to capture context. Recall that we were investigating whether the plurality and definiteness of an NP could be predicted from its context. Therefore, it is plausible to assume that such predictions also benefit from using (contextual) PLMs.

To this end, we fine-tuned PLMs on our dataset. The way we used PLMs is also shown in Figure 4.1, where we fed Chinese characters directly into the PLMs as input and then contacted the hidden representations of the first token and the last token of the target NP together as the representation of the target NP. After that, we still obtained the prediction labels by a linear layer.

In this study, we tried the following PLMs. (1) Chinese BERT and RoBERTa [25,48]. (2) BERT-wwm: vanilla Chinese BERT was pre-trained as a fully character-based model, but [49] proved that the performance can be boosted if Whole Word Masking (WWM; rather than character level masking) is done during pre-training. (3) mBERT: since in addition to Chinese, there are multiple other "cool" languages (e.g., Japanese, Korean and Arabic), we, therefore, wanted to validate whether the predictions can benefit from multilingual pre-training or not.

5 Experiments

In this section, we introduce the settings of the models, the evaluation protocol, and report the performance of the models.

5.1 Settings

In this subsection, we present the results of finding hyperparameters for each model in the preliminary experiments.

Random Forest By finding the hyperparameters, we found that (1) for plurality, the number of decision trees is 1000, the maximum depth of the tree is 500, the minimum number of samples of nodes is 20, the minimum number of samples of leaves is 1, and the n of the N-gram is 3; (2) for definiteness, the number of decision trees is 1000, the maximum depth of the tree is 1000, the minimum number of samples of nodes is 5, the minimum number of samples of leaves is 5, and the n of the N-gram is 4.

Support Vector Machine For SVM, we used radial based function as the kernel function. In addition, by finding the hyperparameters, we found that (1) for plurality, the coefficient of the relaxation factor is 4.05 e+5, and the n of the N-gram is 5; (2) for definiteness, the coefficient of the relaxation factor is 7.22 e+5, and the n of the N-gram is 4.

Logistic Regression For LR, we used Lasso regression to prevent model overfitting. Specifically, by finding the hyperparameters, we found that (1) for plurality, the coefficient of the regularization is 2.01, and the n of the N-gram is 4; (2) for definiteness, the coefficient of the regularization is 2.07, and the n of the N-gram is 5.

Bi-LSTM After finding the hyperparameters, the best performance was achieved for both tasks with a batch size of 512, a hidden size of 128 for the LSTM layer, and a dropout rate of 0. The learning rate was taken as 1e-3 and the number of epochs is 5.

BERT and its Variants For all models in the BERT family, we did not make changes to the model architecture in our experiments; we simply set the dropout rate to 0.1, the learning rate to 1 e-5, and the number of epochs to 5.

5.2 Evaluation Protocol

As introduced in the previous section we randomly select 124K samples from the dataset and obtain the training set, development set, and test set in the ratio of 8:1:1. We tuned the hyper-parameters of each of our models on the development set and chose the setting with the best macro F1 score. We report the macro/weighted averaged precision, recall, and F1 on the test set.

It is worth noting that it has been proposed that the performance of the PLM-based models varies

Table 5 P-values for the McNemar's test for any two models with different random seeds. On the left are the results for the predicted plurality task, and on the right are the results for the predicted definiteness task.

P value	RS = 1	RS = 29	RS = 47	RS = 65	RS = 83
RS = 1	-				
RS = 29	1.0000	-			
RS = 47	.8771	.8988	-		
RS = 65	.1991	.2042	.1663	-	
RS = 83	.7212	.6877	.5586	.3805	-

(Plurality)

P value	RS = 1	RS = 29	RS = 47	RS = 65	RS = 83
RS = 1	-				
RS = 29	.1213	-			
RS = 47	.5341	.3702	-		
RS = 65	.6706	.2771	.8696	-	
RS = 83	.3369	.5768	.7629	.6228	-

(Definiteness)

with the different initialization schemes [50]. To make the experimental results more meaningful, it is of great interest to explore the effect of different random seed values on the performance of the PLM-based models. We took BERT as an example and trained the computational models using five different random ($rs = 1, 29, 47, 65, 83$) seed values for both tasks in this study. For each model, the hyperparameters were kept constant except for the random seeds. Each model was trained for 5 epochs and the checkpoint corresponding to the highest macro F1 score was selected.

Intuitively from the performance of the model with different seed fetches, random seeds have no effect on the model performance (see Appendix B for the confusion matrix for different models). To be more rigorous, for each task, we paired models corresponding to different random seeds in pairs and used McNemar's test[10] [51,52] to explore whether there were statistically significant differences in the performance of the models.

The results are shown in Table 5, where the p-value of McNemar's test for any two models with different random seeds is greater than 0.05, indicating that we accept the hypothesis that there is no statistically significant difference in the predictive performance of these models. In other words, different initialization schemes (different random seed values) have no effect on model performance.

5.3 Experimental Results

Table 6 describes the performance of the computational models predicting the plurality and the definiteness. Generally speaking, the results show that the models can predict the plurality and the definiteness well, which indicates that the models can learn useful information from the context. An interesting point is that the models predicted the definiteness worse than predicting the plurality (the lower weighted scores in definiteness predictions compared to plurality predictions), which was consistent with native Chinese speakers being less sensitive to definiteness than plurality. This suggests that the insensitivity of native Chinese speakers to the definiteness to a greater extent of the characteristics of Chinese itself, namely, that the definiteness of Chinese NPs is contained in the context and that the definitions of definiteness are vague and complex, in addition to other factors

[10] The McNemar's test can be used to check how well the predictions match between one model and another. The assumptions for using McNemar's test are that the observations of the samples should just contain two labels and the samples should be matched-paired. In our research, the observations of the samples have two labels for each task, and secondly, we use different models to predict the same samples, so the assumptions are met.

Table 6 The performance of our models for plurality and definiteness predictions depicted in Section 4. “P”, “R” and “F” stand for precision, recall and F-score respectively. The best results are **boldfaced**, whereas the second best are underlined. The PLMs that do not mark ‘(large)’ use their base version. For many Chinese PLMs, only the base models are publicly available.

	Plurality						Definiteness					
	Macro avg			Weighted avg			Macro avg			Weighted avg		
	P	R	F	P	R	F	P	R	F	P	R	F
RF	81.08	58.19	58.53	80.26	79.69	74.19	68.63	67.24	67.10	68.51	68.09	67.47
LR	76.08	67.39	69.79	80.11	81.58	79.77	71.73	71.53	71.58	71.78	71.82	71.75
SVM	75.56	67.37	69.69	79.88	81.40	79.65	71.34	71.04	71.10	71.37	71.40	71.29
BiLSTM	79.31	70.94	73.59	82.49	83.50	82.14	76.78	76.88	76.80	76.95	76.84	76.87
BERT	80.88	77.96	79.24	85.23	85.73	85.37	81.60	81.66	81.63	81.71	81.69	81.69
BERT—wwm	80.94	**78.34**	**79.50**	**85.38**	85.83	**85.52**	81.95	81.82	81.87	81.98	81.98	81.97
mBERT	80.07	76.96	78.30	84.58	85.15	84.74	80.70	80.41	80.50	80.68	80.66	80.62
RoBERTa	81.21	77.53	79.09	85.22	85.79	85.35	**82.27**	**82.10**	**82.16**	**82.28**	**82.28**	**82.26**
RoBERTa (large)	**81.72**	77.37	79.17	**85.38**	**85.98**	85.46	81.80	81.58	81.66	81.79	81.79	81.76

(e.g. educational attainment of different people).

In terms of model performance, as expected Bi-LSTM and PLM-based models outperformed ML-based models. As a baseline ML-based model, LR performed best, achieving weighted-averaged F-scores of 79.77 for plurality predictions and 71.75 for definiteness predictions. Bi-LSTM with Glove embeddings defeated all ML-based models but lost to PLM-based models. This embodies that context plays an important role in the prediction of plurality and definiteness, which is consistent with the definition of “cool” (see Section 1). Moreover, it also shows that PLM-based models capture context better than Bi-LSTM, and PLMs embed Chinese better than Glove.

Regarding the BERT-based model, we have the following findings: (1) BERT-wwm performed remarkably well. It generally performed the best for plurality prediction and was the second-best model for definiteness prediction. This demonstrated that, on our task, BERT does benefit from whole word mask pre-training probably because the intended meaning of a word (noun in our situation) is mainly inferred from its context rather than its inner structure. (2) BERT did not benefit from multilingual pre-training since mBERT received 84.74 weighted F-score on plurality predictions and 80.62 on definiteness predictions though mBERT was pre-trained on typical “cool” languages, including Arabic, Japanese, and Korean. This is consistent with the fact that mBERT does not perform as well as BERT in other downstream tasks. This probably attributes to the fact that speakers of these “cool” languages use contexts differently and, therefore, multi-lingual pre-training may not yield substantial benefits to downstream tasks that rely on context. In other words, in the tasks of this study, other “cool” languages are a kind of noise rather than a useful auxiliary material for Chinese. This makes supervision signals become needed. In the future, it would be valuable to build an NP corpus in multiple “cool” languages and see whether the predictions can benefit or not. (3) On our tasks, the amount of parameters is not the more the better. RoBERTa (large) performed worse than the BERT-wwm on plurality predictions and worse than RoBERTa (base) on definiteness predictions. Further probing experiments are needed to explain what happens. (4) Last but not least, the Roberta-based models achieved good results on two tasks. Although RoBERTa (large) did not achieve the best results as expected, it also got the second-highest weighted F-score for plurality prediction. On the other hand, RoBERTa (base) was the best model in predicting definiteness. This indicates that the RoBERTa dynamic masking policy is helpful for both tasks. Dynamic masks make RoBERTa pre-trained data masks more flexible and the hidden representations contain more contextual information so that RoBERTa-based models are more capable of capturing the context.

6 Analysis

We also carried out further analyses of the model, which included explanations of the performance of models as well as three post-hoc analyses. We described them in detail below this section.

6.1 The Explainability of Computational Models

As mentioned in Section 2, we chose LR and RF because they both have better model explainability. In addition, we can try to explain the performance of PLM-based models with the help of the attention weights. In the following, we try to explain the performance of the LR, RF, and PLM-based models.

6.1.1 Explanation of LR performance

Although the LR was the best-performing of all ML-based models, the features it considered most important are completely inconsistent with human intuition. Figure 6.1 shows the 30 features that LR considered most important for prediction plurality (15 of which were most important for prediction plural and 15 for prediction singular), and Figure 6.2 shows the results of the top 30 most valuable features corresponding to prediction definiteness (15 of which were most important for prediction definite and 15 for prediction indefinite). In our intuition, the most important features for prediction plurality should be quantifiers, plural markers, etc., but none of them appears in the features selected by LR, and this situation also appears in prediction definiteness. This interesting phenomenon is one of the elements that we need to explore further in our future work.

6.1.2 Explanation of RF performance

Unlike LR, RF selected features that are more in line with human intuition. Figure 6.3 and Figure 6.4 show the top 30 most important features for RF to complete the prediction plurality and definiteness.

For plurality, the first four important features are “那些 (nà xiē)”, “np 这些 (zhè xiē)”, “这些 (zhè xiē)” and “np 那些 (nà xiē)” (where “np” is the Chinese NP position marker, see Section 4). This suggests that RF considers demonstratives to be important for predicting plurality. Although, in the previous analysis, we showed that “们 (men)” may not be a plural marker, RF believes that “们 (men)” is also very helpful in predicting plurality (8 out of the first 30 features have “们 (men)”). In addition, there are some words that can be used directly in response to plurality that are also considered very important, such as “自己 (zì jǐ)” (*me*) and ” 父母 (fù mǔ)” (*parents*).

For definiteness, RF continues to believe that definite markers are the most important, such as, “这个 (zhè gè)” (*this*), “那个 (nà gè)” (*that*), “这些 (zhè xiē)” (*these*) and “我的 (wǒ de)” (*my*). However, RF also chose some features with unclear meanings, such as “蚊子 (wén zi)” (*mosquitoes*), in predicting definiteness as LR did.

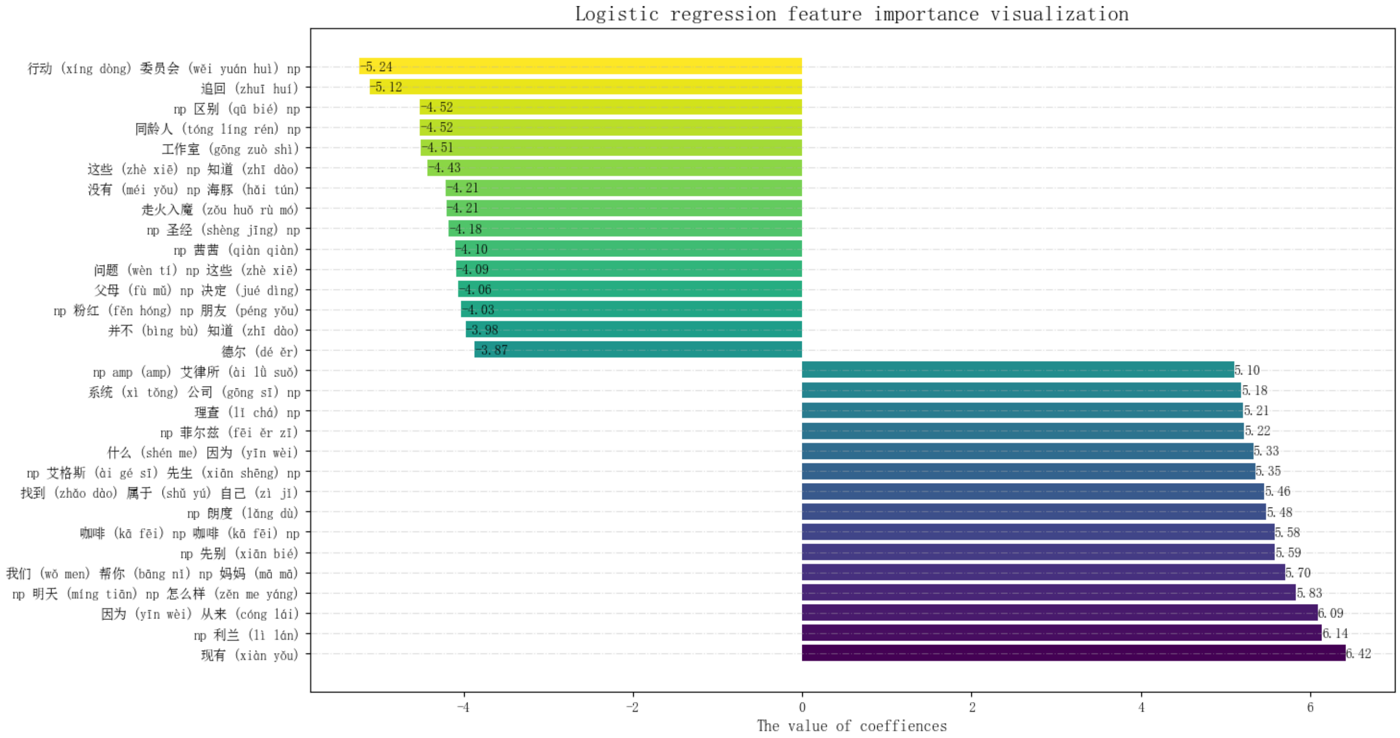


Figure 6.1 The 30 features that LR considered most important for prediction plurality (15 of which were most important for prediction plural and 15 for prediction singular).

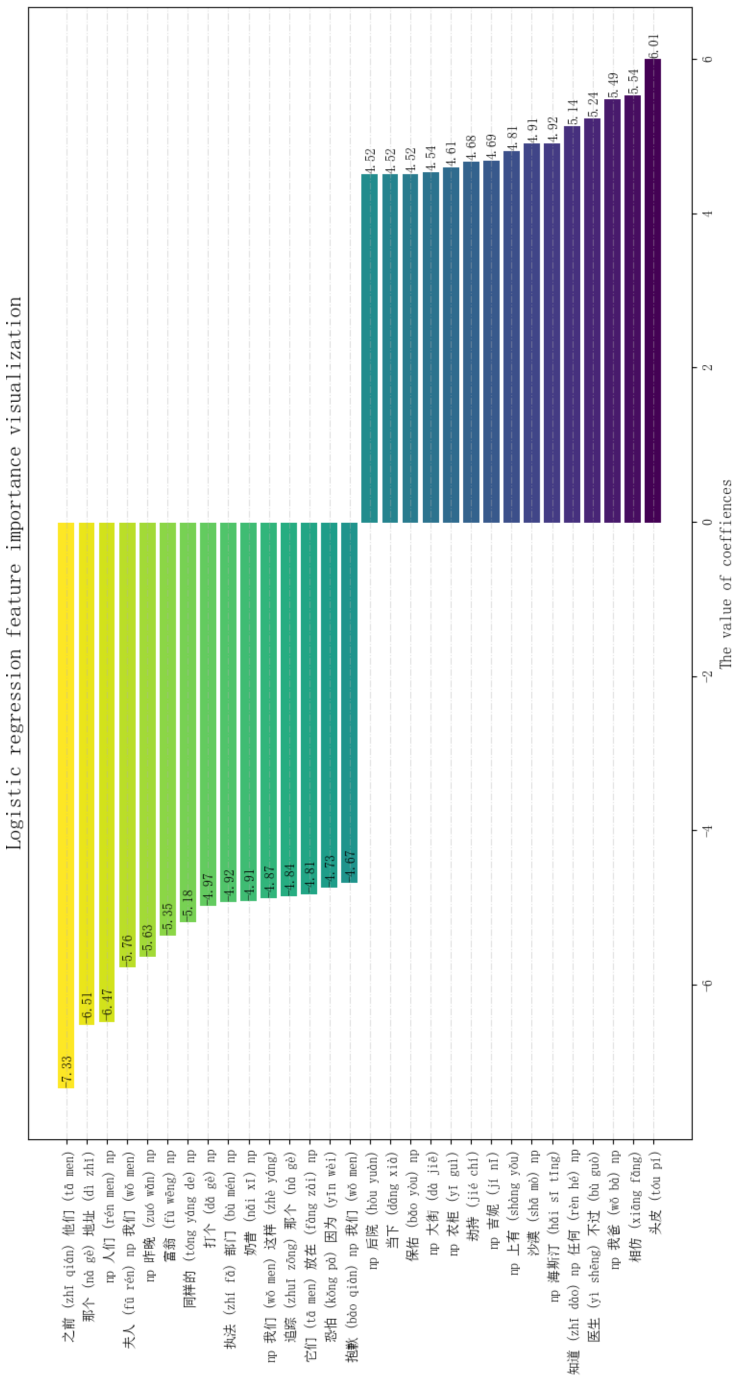


Figure 6.2 The 30 features that LR considered most important for prediction definiteness (15 of which were most important for prediction definite and 15 for prediction indefinite).

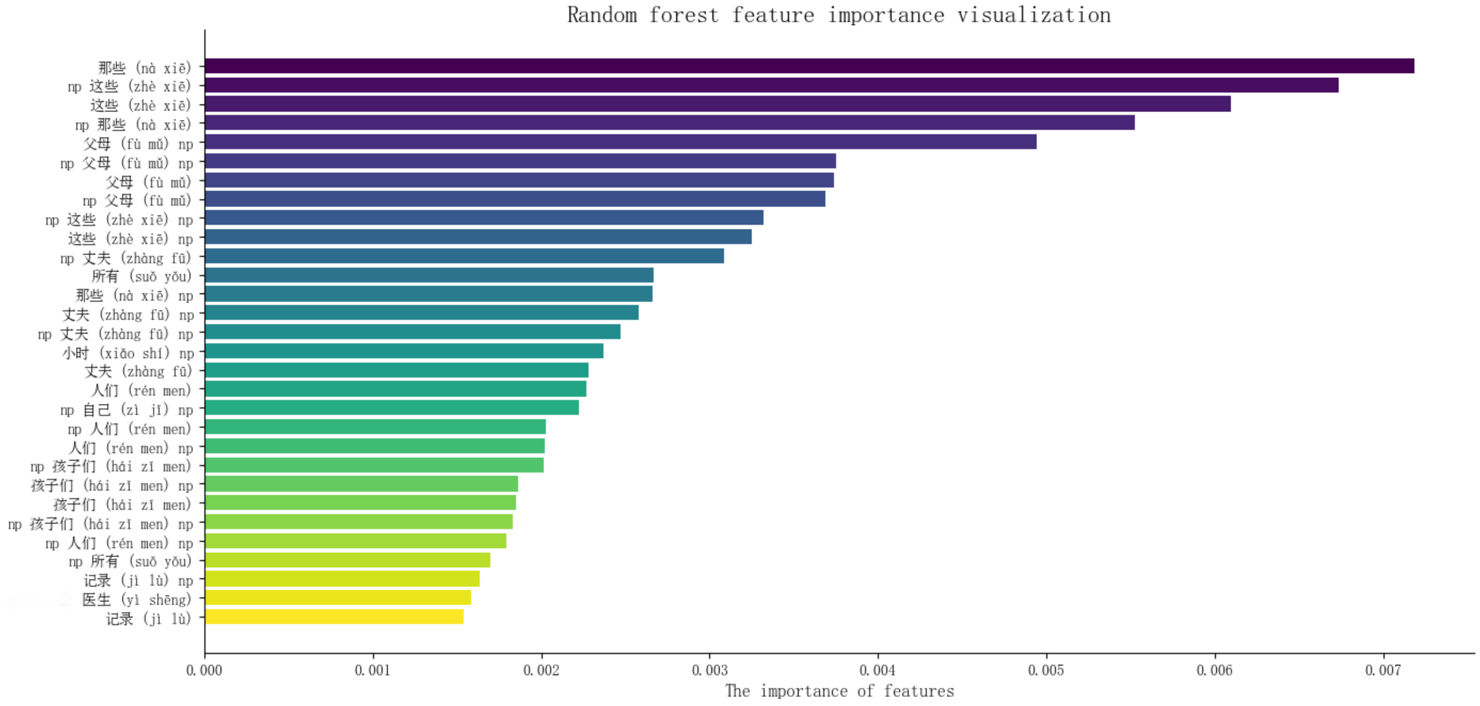


Figure 6.3 The 30 features that RF considered most important for prediction plurality.

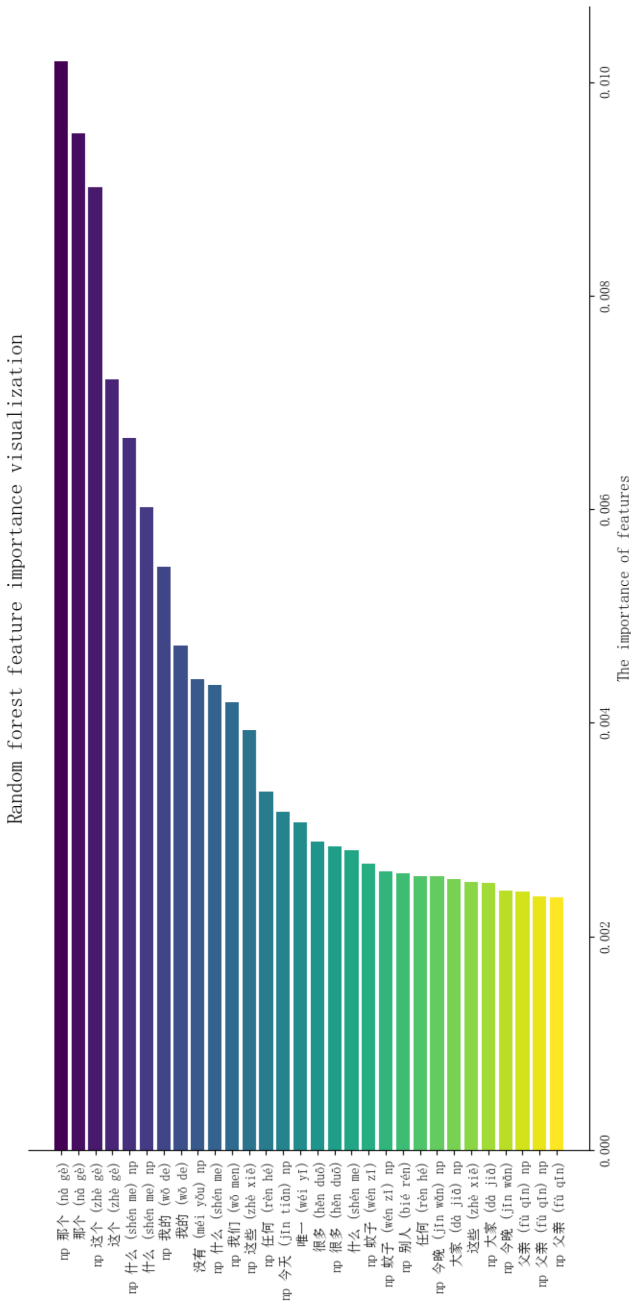


Figure 6.4 The 30 features that RF considered most important for prediction definiteness.

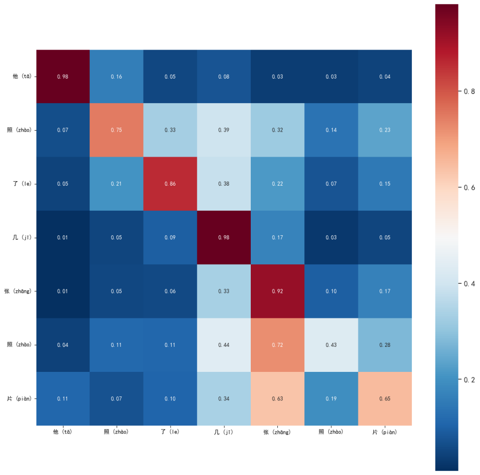


Figure 6.5 Example 1 of heatmap for prediction plurality

6.1.3 Explanation of BERT performance

We used BERT as an example to explain the predictions of PLM-based models using attention-matrix heatmaps [53] to visualize attention weights. It needs to be stated that the heatmaps shown below are the results of summing and normalizing the attention weights of all heads after removing the special markers of BERT (“[CLS]” and “[SEP]”).

(6) a. 他 照 了 * 几张 照片 *。
tā zhào le jǐzhāng zhàopiàn
He took a few pictures.

Figure 6.5 shows the results of the heatmap for predicting plurality of “几张 (jǐ zhāng) 照片 (zhào piān)” (*a few photos*) in the sentence (example as shown in (6-a)). The heatmap shows that “几张 (jǐ zhāng)” has a greater influence on “照片 (zhào piàn)” (The value of “几 (jǐ)” on “照 (zhào)” is 0.44, “几 (jǐ)” on “片 (piàn)” is 0.34, “张 (zhāng)” on “照 (zhào)” is 0.72, and “张 (zhāng)” on “片 (piàn)” is 0.63). In addition, the effect of “几 (jǐ)” on “张 (zhāng)” is also large (is 0.33), suggesting that the model is aware that it is “张 (zhāng)” preceded by “几 (jǐ)” rather than “一 (yī)”, and that it refers to several photos rather than one. This indicates that the model successfully inferred that there were multiple photos.

(7) a. 所以 是 有 * 两 个 不同 的 人 *。
suǒyǐ shì yǒu liǎng gè bùtóng de rén
So there are two different people.

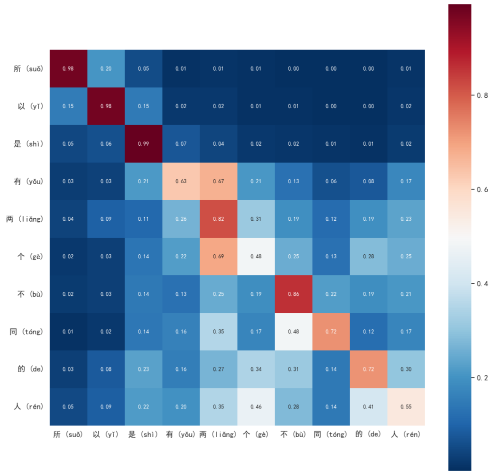


Figure 6.6 Example 2 of heatmap for prediction plurality

Figure 6.6 shows another example (as shown in (7-a)) of prediction plurality. From the heatmap we can see that “两 (liǎng)” and “个 (gè)” have the greatest influence on the noun “人 (rén)” compared to the other tokens except itself and the previous token, the values are 0.35 and 0.46 respectively. “两 (liǎng)” is a quantifier and “个 (gè)” is an article, which indicates that for BERT quantifiers and articles help a lot in predicting plurality, which is consistent with human intuition.

(8) a. 我 看到 * 我 这 一代 最 杰出 的 人 * 。
wǒ kàndào wǒ zhè yídài zuì jiéchū de rén
I see the most outstanding people of my generation.

The same happens for BERT definiteness prediction. Figure 6.7 shows the heatmap of attention weights of BERT predicting definiteness (the sentence of example is shown in (8-a)). From the figure, we can see that, in addition to itself, there are “我 (wǒ)”, “这 (zhè)” and “最 (zuì)” that have a relatively large effect on “一代 (yí dài)”. This indicates that the model understands that this refers to “*my generation*” rather than “*a generation*”.

(9) a. * 这 辆 车 * 自动 驾驶 是否 合法 呢?
zhè liàng chē zìdòng jiàshǐ shìfǒu héfǎ ni
Is it legal for this car to drive itself?

Figure 6.8 shows another example of prediction definiteness (as shown in (9-a)). As well, apart from “车 (chē)” itself, “这 (zhè)” and “辆 (liàng)” have the greatest influence on it. This indicates that the model notices the definite marker and assumes that “车 (chē)” here refers to a specific car.

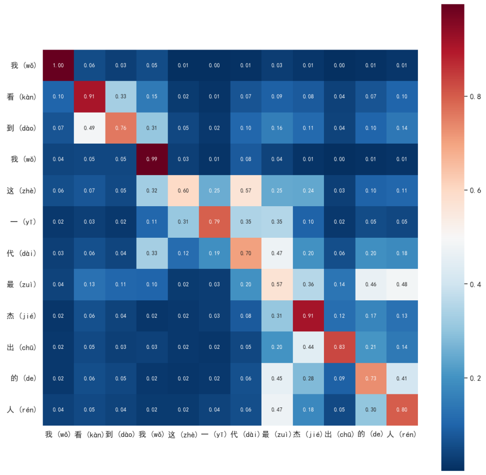


Figure 6.7 Example 1 of heatmap for prediction definiteness

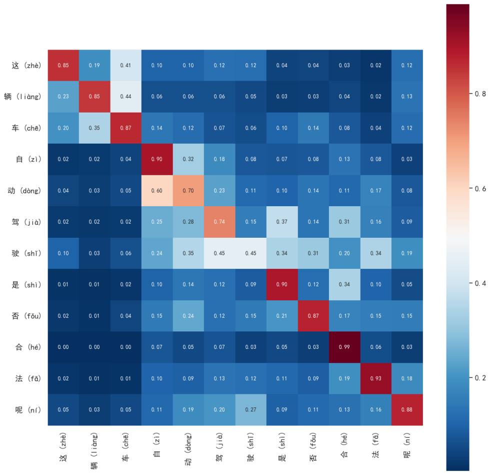


Figure 6.8 Example 2 of heatmap for prediction definiteness

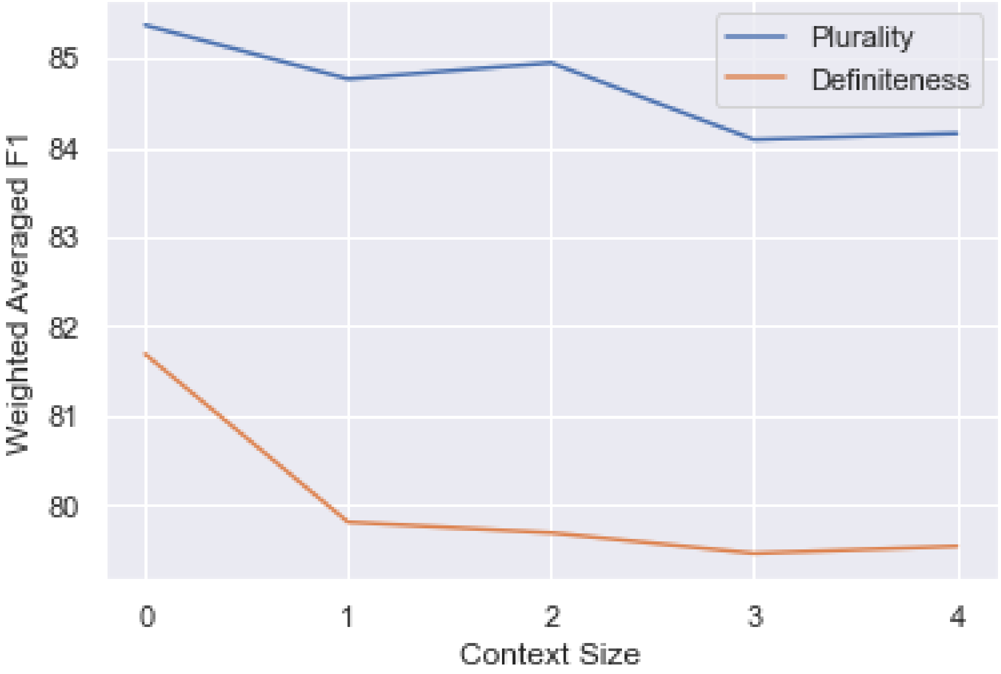


Figure 6.9 Weighted F1 concerning different context sizes. The size is measured by the number of sentences around the target sentence.

In general, through the analysis of the above four examples we can easily find that BERT's prediction of plurality and definiteness is in line with our intuition and it has nicely explainability. Meanwhile, the important features selected by the RF are also explainable. But we cannot give a reasonable explanation for the prediction of LR, which needs further probing.

6.2 Post-hoc Analyses

In this subsection, we further explore three interesting and meaningful questions based on the computational models: (1) What is the effect of Context Size? (2) Do the plurality and definiteness predictions help each other? (3) How does the explicitness impact model's behaviors?

6.2.1 What is the impact of Context Size?

According to what linguists hypothesized [1], the interpretation of the plurality and definiteness of an NP relies on its context and such context is not necessarily only the current sentence but also the whole discourse. For example, without more context, it is hard to decide the plurality of the NP in example (1-c). However, in the current experimental setting, we only fed the models with only one sentence, namely the target sentence.

Therefore, it is plausible to expect that if we increase the size of contexts, the predictions become

Table 7 The results of 4-way prediction and the merged results of 2 binary predictions.

	4-way						2-way (merged)					
	Macro avg			Weighted avg			Macro avg			Weighted avg		
	P	R	F	P	R	F	P	R	F	P	R	F
BERT	67.37	64.26	65.53	70.72	71.20	70.79	65.62	63.35	64.34	69.49	69.91	69.61
BERT—wwm	67.94	65.74	66.72	71.54	71.86	71.62	66.51	**64.23**	**65.24**	70.03	70.40	70.14
mBERT	67.73	64.58	65.69	71.12	71.46	71.01	64.19	61.51	62.62	68.11	68.59	68.21
RoBERTa	68.25	**66.42**	**67.24**	72.03	72.36	72.14	67.08	63.89	65.23	**70.29**	**70.74**	**70.36**
RoBERTa (large)	**68.73**	65.51	66.87	**72.09**	**72.55**	**72.18**	**67.11**	63.36	64.90	69.90	70.35	69.92

more accurate. To test this idea, we increased the size of the contexts and evaluated the performance of BERT with different context lengths as inputs. The results are shown in Figure 6.9. We used the weighted average F-score to evaluate the performance of models. We used 0-5 to denote different context lengths, e.g., 1 means combining the target sentence and each of the consecutive sentences before and after it as one input data. Due to the limitation of a maximum of 512 tokens of BERT input data, we combined at most 4 consecutive sentences before and after each target sentence with the target sentence as training data.

Nevertheless, different from the expectation, the performance of both tasks decreases with the increase of the context size. The decrease in performance is more pronounced in prediction definiteness compared to prediction plurality. A possible explanation is that although wider contexts add useful information to the prediction, it also adds confusion as our focus is only a small part (i.e., the NP) of the target sentence. This makes it hard for the model to extract useful information from the representation of a wide context, and add it to the representation of the target NP (which is often a few words; recall that we only used the representation of the target NP for prediction), and make predictions. It is worth noting that similar phenomena are observed in other pragmatics tasks [54, 55].

6.2.2 Do the plurality and definiteness predictions help each other?

Since both plurality and definiteness are information carried by NPs. One could expect that the information that is needed for predicting the plurality of an NP might help determine the definiteness of the same NP and vice versa. In other words, we might benefit from predicting plurality and definiteness simultaneously. Rather than employing multi-task learning, we opted to fine-tune the models for 4-way predictions. Specifically, given an NP, the models classify it into one of four categories: indefinite singular, indefinite plural, definite singular, or definite plural. To fairly compare the model performance for 4-way prediction and 2 separate binary predictions, we merged the predictions obtained in Section 5 and re-computed each score.

Table 7 reports the performance of each model on 4-way and merged binary predictions. The results suggest that models can significantly benefit from predicting plurality and definiteness simultaneously compared to predicting them separately. For example, in joint prediction, RoBERTa

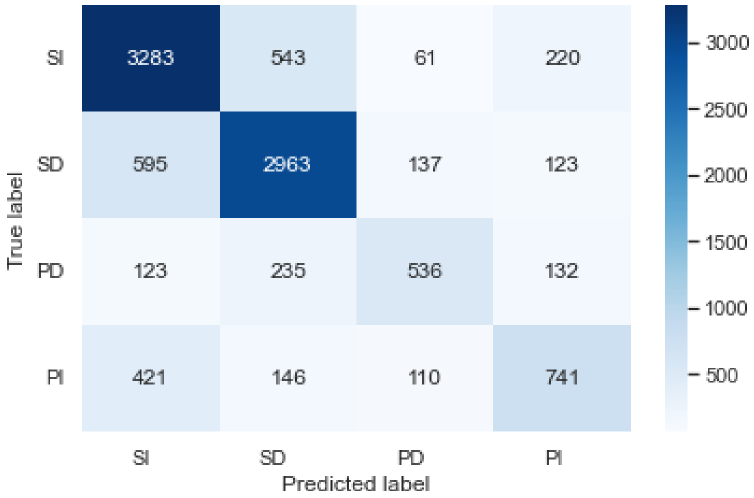


Figure 6.10 The confusion matrix for 4-way prediction of RoBERTa (large), in which S, P, I, and D mean "singular", "plural", "indefinite" and "definite", respectively.

achieved a weighted average F1 score of 72.14. However, when doing binary predictions, the merged weighted F1 score dropped to 70.36.

Focusing on the 4-way prediction results, we found that akin to the binary predictions, RoBERTa had the best performance. It achieved a weighted F1 score of 72.14 and a macro F1 score of 67.24. It was followed by RoBERTa (large), which had an on-par weighted F1 and lower micro F1. BERT-wwm performed slightly worse than them, but still remarkably well. Figure 6.10 is the confusion matrix of Roberta-large for joint prediction (see Appendix C for more confusion matrices for joint prediction), which further ascertains the theory that deciding definiteness is hard in Chinese as although the labels of the plurality are way more imbalanced than that of the definiteness (see Table 1) the model is still much easier to confuse between "definite" and "indefinite" than between "singular" and "plural".

6.2.3 How does the explicitness impact model's behaviours?

In the corpus analysis, we identified that NPs in 12.42% and 15.86% samples from our dataset explicitly express plurality and definiteness respectively. Since these explicit expressions provide clear markers, we expected that the predictions of both tasks on explicit expressions are easier than on implicit expressions. Thus, models would receive higher scores on the portion of explicit expressions. To examine this, we assessed BERT-based models on implicit and explicit expressions respectively and report the results in Figure 6.11[11]. As expected, for both tasks, all models performed better on explicit expressions than implicit expressions.

Besides, we also have some interesting observations: (1) the difference between the performance on explicit expressions and on implicit expressions is larger on plurality prediction than definiteness

[11] To highlight the differences, we report macro-F this time.

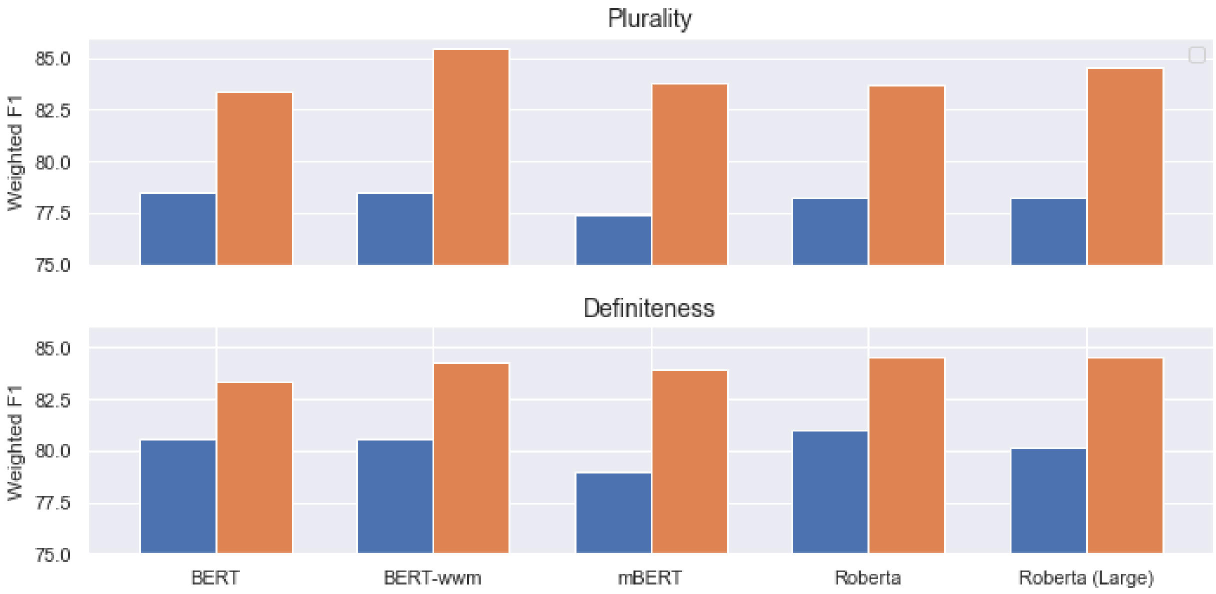


Figure 6.11 Macro F-scores of BERT-based models on implicit and explicit expressions of plurality and definiteness. The blue bars indicate the performance of models on implicit expressions while the orange bars indicate that on explicit expressions.

prediction. (2) For plurality prediction, except mBERT, all other models have similar performance on implicit expressions. BERT-wwm performed significantly better on explicit expressions than other models. (3) For definiteness prediction, RoBERTa performed the best on both implicit and explicit expressions.

7 Conclusion

In this section, we first give a discussion of the work in this study, and then list the future work.

7.1 Discussion

In the current paper, we studied the comprehension of plurality and definiteness in Chinese NPs, which, theoretically, can be implicitly expressed in Chinese because their meaning can be inferred from the contexts of NPs. In this study, two research questions were posed from both comprehension and computational perspectives.

From the comprehension perspective, we were interested in *How frequently are plurality and definiteness omitted in Chinese?* To check how frequently they are expressed implicitly, we first constructed a Chinese corpus where NPs are marked with their plurality and definiteness. Two assessment studies showed that our dataset is of good quality. A corpus analysis suggests that Chinese speaker drops plural and definiteness markers very frequently. The rate of the plurality omitted in the dataset is 87.58%, and the rate corresponding to the definiteness is 84.14%.

From the computational perspectives, we wanted to use computational models to understand *How well are neural models (which are thought to capture context quite well) able to predict information about plurality/definiteness in Chinese Noun Phrases?* Based on the dataset, we have built computational models using the classical ML-based models, Bi-LSTM, and the latest PLM-based models. The experimental results suggest that the meaning of plurality and definiteness is indeed predictable. Looking at the difference between the model's predictions for plurality and definiteness, the model generally predicted plurality better than definiteness. In terms of the comparison of the performance of ML-based models, Bi-LSTM, and PLM-based models, we found that the PLM-based models performed best for plurality prediction and definiteness prediction, with the weighted F-score of plurality prediction higher than 84, and the weighted F-score of definiteness prediction higher than 80. Moreover, for plurality prediction, BERT-wwm performed best, with its weighted F-score up to 85.52. For definiteness prediction, RoBERTa performed best, with its weighted F-score up to 82.26. Bi-LSTM model performed between ML-based models and PLM-models, the weighted F-score for plurality prediction is 82.14, and for definiteness prediction is 76.87. ML-based models did not perform as well as the first two, with LR and SVM getting weighted F-scores of around 79 for plurality prediction and 71 for definiteness prediction. The worst performer is RF, the weighted F-score for plurality prediction is 74.19, and for definiteness prediction is 67.47.

7.2 Future Work

Meanwhile, there is still a lot of room for improvement in our research, and we will continue to work mainly in these areas in the future:

(1) As the fact that in English there also are cases where it's not easy to know whether a noun phrase

is definite. In the future, we will compare the differences between English and Chinese in terms of expressing definiteness implicitly.

(2) In this study, we explored the performance of computational models to predict the plurality and definiteness of Chinese NPs. But we did not explore how well people perform in predicting plurality and definiteness of Chinese NPs. It is interesting to explore human performance and compare it to computational models. This is one of our future work.

(3) Our corpus analyses and computational modeling were done on the data from a single source, namely, conversations in TV episodes. It is not fully clear whether our findings can be generalized to data in other genres. In the future, we will try more different types of corpus to further expand our dataset.

(4) In this research, we investigate whether linguists' understanding of "们 (men)" can be verified on the dataset. In the future we intend to go further and validate more interesting perspectives of linguists on the dataset, such as the one mentioned in Section 2, that native speakers of Chinese have a habit that placing Chinese NPs in sentence-initial positions to indicate definiteness.

(5) In making interpretations of the computational model performance, we found that the features that LR considered important differed significantly from human intuition. We will explore further in the future what exactly happened here.

(6) Last but not least, our dataset did not annotate disagreements, our models also do not have such an ability. In the future, we plan to explore the disagreements in the comprehension of not only the definiteness and plurality of Chinese NPs but also other components in Chinese that are omittable.

References

[1] C-T James Huang. On the distribution and reference of empty pronouns. *Linguistic inquiry*, pages 531–574, 1984.

[2] Kees van Deemter. Dimensions of explanatory value in nlp models. *Computational Linguistics*, pages 1–20.

[3] Fahime Same, Guanyi Chen, and Kees Van Deemter. Non-neural models matter: a re-evaluation of neural referring expression generation systems. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5554–5567, Dublin, Ireland, May 2022. Association for Computational Linguistics.

[4] Guanyi Chen, Kees van Deemter, and Chenghua Lin. Generating quantified descriptions of abstract visual scenes. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 529–539, Tokyo, Japan, October–November 2019. Association for Computational Linguistics.

[5] Herbert P Grice. Logic and conversation. In *Speech acts*, pages 41–58. Brill, 1975.

[6] Richard Newnham. *About Chinese*. Penguin Books Ltd, 1971.

[7] C-T James Huang, Y-H Audrey Li, and Yafei Li. The syntax of chinese. *(No Title)*, 2009.

[8] Guanyi Chen, Kees van Deemter, and Chenghua Lin. Modelling pro-drop with the rational speech acts model. In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 159–164, Tilburg University, The Netherlands, November 2018. Association for Computational Linguistics.

[9] Guanyi Chen and Kees van Deemter. Lessons from computational modelling of reference production in Mandarin and English. In *Proceedings of the 13th International Conference on Natural Language Generation*, pages 263–272, Dublin, Ireland, December 2020. Association for Computational Linguistics.

[10] Guanyi Chen and Kees van Deemter. Understanding the use of quantifiers in Mandarin. In *Findings of the Association for Computational Linguistics: AACL-IJCNLP 2022*, pages 73–80, Online only, November 2022. Association for Computational Linguistics.

[11] Guanyi Chen. *Computational generation of Chinese noun phrases*. PhD thesis, Utrecht University, 2022.

[12] Guanyi Chen, Fahime Same, and Kees van Deemter. Neural referential form selection: Generalisability and interpretability. *Computer Speech & Language*, 79:101466, 2023.

[13] Stephanie Schoch, Diyi Yang, and Yangfeng Ji. “this is a problem, don’t you agree?” framing and bias in human evaluation for natural language generation. In *Proceedings of the 1st Workshop on Evaluating NLG Evaluation*, pages 10–16, 2020.

[14] Johannes Fürnkranz. A study using n-gram features for text categorization. *Austrian Research Institute for Artifical Intelligence*, 3(1998):1–10, 1998.

[15] Marti A. Hearst, Susan T Dumais, Edgar Osuna, John Platt, and Bernhard Scholkopf. Support vector machines. *IEEE Intelligent Systems and their applications*, 13(4):18–28, 1998.

[16] Raymond E Wright. Logistic regression. 1995.

[17] Larry R Medsker and LC Jain. Recurrent neural networks. *Design and Applications*, 5:64–67, 2001.

[18] Klaus Greff, Rupesh K Srivastava, Jan Koutník, Bas R Steunebrink, and Jürgen Schmidhuber. Lstm: A search space odyssey. *IEEE transactions on neural networks and learning systems*, 28(10):2222–2232, 2016.

[19] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.

[20] Mike Schuster and Kuldip K Paliwal. Bidirectional recurrent neural networks. *IEEE transactions on Signal Processing*, 45(11):2673–2681, 1997.

[21] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[22] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[23] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[24] Justyna Sarzynska-Wawer, Aleksander Wawer, Aleksandra Pawlak, Julia Szymanowska, Izabela Stefaniak, Michal Jarkiewicz, and Lukasz Okruszek. Detecting formal thought disorder by deep contextualized word representations. *Psychiatry Research*, 304:114135, 2021.

[25] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[26] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018.

[27] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[28] Leo Breiman. Random forests. *Machine learning*, 45:5–32, 2001.

[29] Shivam Agarwal. Data mining: Data mining concepts and techniques. In *2013 international conference on machine intelligence and research advancement*, pages 203–207. IEEE, 2013.

[30] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*, 2016.

[31] Michael Alexander Kirkwood Halliday and Ruqaiya Hasan. *Cohesion in english*. Routledge, 2014.

[32] 胡壮麟. 国外汉英对比研究杂谈 (二, 续完). 语言教学与研究, (2):117–128, 1982.

[33] Daniel Robertson. Variability in the use of the english article system by chinese learners of english. *Second language research*, 16(2):135–172, 2000.

[34] 郭鸿杰 and 周芹芹. 中国英语学习者不定冠词习得的变异研究. 解放军外国语学院学报, (1):54–58, 2012.

[35] David Bremmers, Jianan Liu, Martijn van der Klis, and Bert Le Bruyn. Translation mining: Definiteness across languages (a reply to jenks 2018). *Linguistic Inquiry*, 53(4):735–752, 2022.

[36] Elaine Tarone and Betsy Parrish. Task-related variation in interlanguage: The case of articles. *Language learning*, 38(1):21–44, 1988.

[37] 邢辐义. 论"们"和"诸位"之类并用. 中国语文, (6):289–289, 1960.

[38] Robert Iljic. Quantification in mandarin chinese: Two markers of plurality. 1994.

[39] Yen-hui Audrey Li. Plurality in a classifier language. *Journal of East Asian Linguistics*, pages 75–99, 1999.

[40] Longyue Wang, Zhaopeng Tu, Shuming Shi, Tong Zhang, Yvette Graham, and Qun Liu. Translating pro-drop languages with reconstruction models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

[41] Franz Josef Och and Hermann Ney. A systematic comparison of various statistical alignment models. *Computational linguistics*, 29(1):19–51, 2003.

[42] Christopher D Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky. The stanford corenlp natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, pages 55–60, 2014.

[43] Jacob Cohen. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46, 1960.

[44] Robert L Brennan and Dale J Prediger. Coefficient kappa: Some uses, misuses, and alternatives. *Educational and psychological measurement*, 41(3):687–699, 1981.

[45] DK Donker, A Hasman, and HP Van Geijn. Interpretation of low kappa values. *International journal of bio-medical computing*, 33(1):55–64, 1993.

[46] Malcolm Maclure and Walter C Willett. Misinterpretation and misuse of the kappa statistic. *American journal of epidemiology*, 126(2):161–169, 1987.

[47] Massimo Poesio, Jon Chamberlain, Silviu Paun, Juntao Yu, Alexandra Uma, and Udo Kruschwitz. A crowdsourced corpus of multiple judgments and disagreement on anaphoric interpretation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1778–1789, 2019.

[48] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[49] Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, and Ziqing Yang. Pre-training with whole word masking for chinese bert. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:3504–3514, 2021.

[50] Thibault Sellam, Steve Yadlowsky, Jason Wei, Naomi Saphra, Alexander D'Amour, Tal Linzen, Jasmijn Bastings, Iulia Turc, Jacob Eisenstein, Dipanjan Das, et al. The multiberts: Bert reproductions for robustness analysis. *arXiv preprint arXiv:2106.16163*, 2021.

[51] Quinn McNemar. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157, 1947.

[52] Thomas G Dietterich. Approximate statistical tests for comparing supervised classification learning algorithms. *Neural computation*, 10(7):1895–1923, 1998.

[53] Alexander M Rush, Sumit Chopra, and Jason Weston. A neural attention model for abstractive sentence summarization. *arXiv preprint arXiv:1509.00685*, 2015.

[54] Mandar Joshi, Omer Levy, Daniel S Weld, and Luke Zettlemoyer. Bert for coreference resolution: Baselines and analysis. *arXiv preprint arXiv:1908.09091*, 2019.

[55] Arup Baruah, Kaushik Das, Ferdous Barbhuiya, and Kuntal Dey. Context-aware sarcasm detection using bert. In *Proceedings of the Second Workshop on Figurative Language Processing*, pages 83–87, 2020.

A Dataset Assessment Appendix

In the assessment, to ensure that each sample was assessed by at least two annotators, we created two sets of questionnaires in each assessment and asked two annotators to respond to each set. Below are specific statistics of the evaluation results, where Table 8, Table 9, and Table 10 are the results of Assessment 1 in which we asked the annotators to assess the labels with yes or no; Table 11 and Table 12 are the results of the assessment by translating yes or no into specific labels; Table 13 and Table 14 are the results of Assessment 2 in which the labels were annotated by annotators themselves. The assessment results presented in Section 3 are the result of aggregating the data from the two questionnaires for each assessment (matrix addition of the statistics from the two questionnaires, then calculate the corresponding $\mathrm{Acc}_{=2}$, $\mathrm{Acc}_{\geq 1}$, and IAAs).

Table 8 Human assessment results of Assessment 1 on NP identification, in which A, B, C, and D represent four annotators.

		B				D	
		True	False			True	False
A	True	170	6	C	True	154	16
	False	21	3		False	22	8
IAA(%)=0.8650				IAA(%)=0.8100			
(Assessment 1 Set 1)				(Assessment 1 Set 2)			

Table 9 Human assessment results of Assessment 1 on plurality annotation, in which A, B, C, and D represent four annotators.

		B				D	
		True	False			True	False
A	True	172	15	C	True	164	20
	False	5	8		False	11	5
IAA(%)=0.9000				IAA(%)=0.8450			
(Assessment 1 Set 1)				(Assessment 1 Set 2)			

A DATASET ASSESSMENT APPENDIX

Table 10 Human assessment results of Assessment 1 on definiteness annotation, in which A, B, C, and D represent four annotators.

		B				D	
		True	False			True	False
A	True	170	6	C	True	154	16
	False	21	3		False	22	8
IAA(%)=0.8650				IAA(%)=0.8100			
(Assessment 1 Set 1)				(Assessment 1 Set 2)			

Table 11 Human assessment results of Assessment 1 on plurality annotation (translate yes or no to the corresponding label) , in which A, B, C, and D represent four annotators.

		B				D	
		Singular	Plural			Singular	Plural
A	Singular	128	23	C	Singular	129	24
	Plural	4	45		Plural	7	40
IAA (κ)=0.6773				IAA (κ)=0.6169			
(Assessment 1 Set 1)				(Assessment 1 Set 2)			

Table 12 Human assessment results of Assessment 1 on definiteness annotation (translate yes or no to the corresponding label) , in which A, B, C, and D represent four annotators.

		B				D	
		Definite	Indefinite			Definite	Indefinite
A	Definite	128	23	C	Definite	129	24
	Indefinite	4	45		Indefinite	7	40
IAA (κ)=0.7251				IAA (κ)=0.6200			
(Assessment 1 Set 1)				(Assessment 1 Set 2)			

A DATASET ASSESSMENT APPENDIX

Table 13 Human assessment results of Assessment 2 on plurality annotation (ask annotators to annotate the labels by themselves) , in which A, B, C, and D represent four annotators.

		B				D	
		Singular	Plural			Singular	Plural
A	Singular	78	3	C	Singular	70	7
	Plural	6	12		Plural	5	17
		IAA (%)=0.9091				IAA (%)=0.8788	
		IAA (κ)=0.6733				IAA(κ)=0.6604	
		(Assessment 2 Set 1)				(Assessment 2 Set 2)	

Table 14 Human assessment results of Assessment 2 on definiteness annotation (ask annotators to annotate the labels by themselves) , in which A, B, C, and D represent four annotators.

		B				D	
		Definite	Indefinite			Definite	Indefinite
A	Definite	51	13	C	Definite	55	10
	Indefinite	13	22		Indefinite	11	23
		IAA(%)=0.7374				IAA(%)=0.7879	
		IAA(κ)=0.4254				IAA(κ)=0.5263	
		(Assessment 2 Set 1)				(Assessment 2 Set 2)	

B Initialization Schemes Experiment Appendix

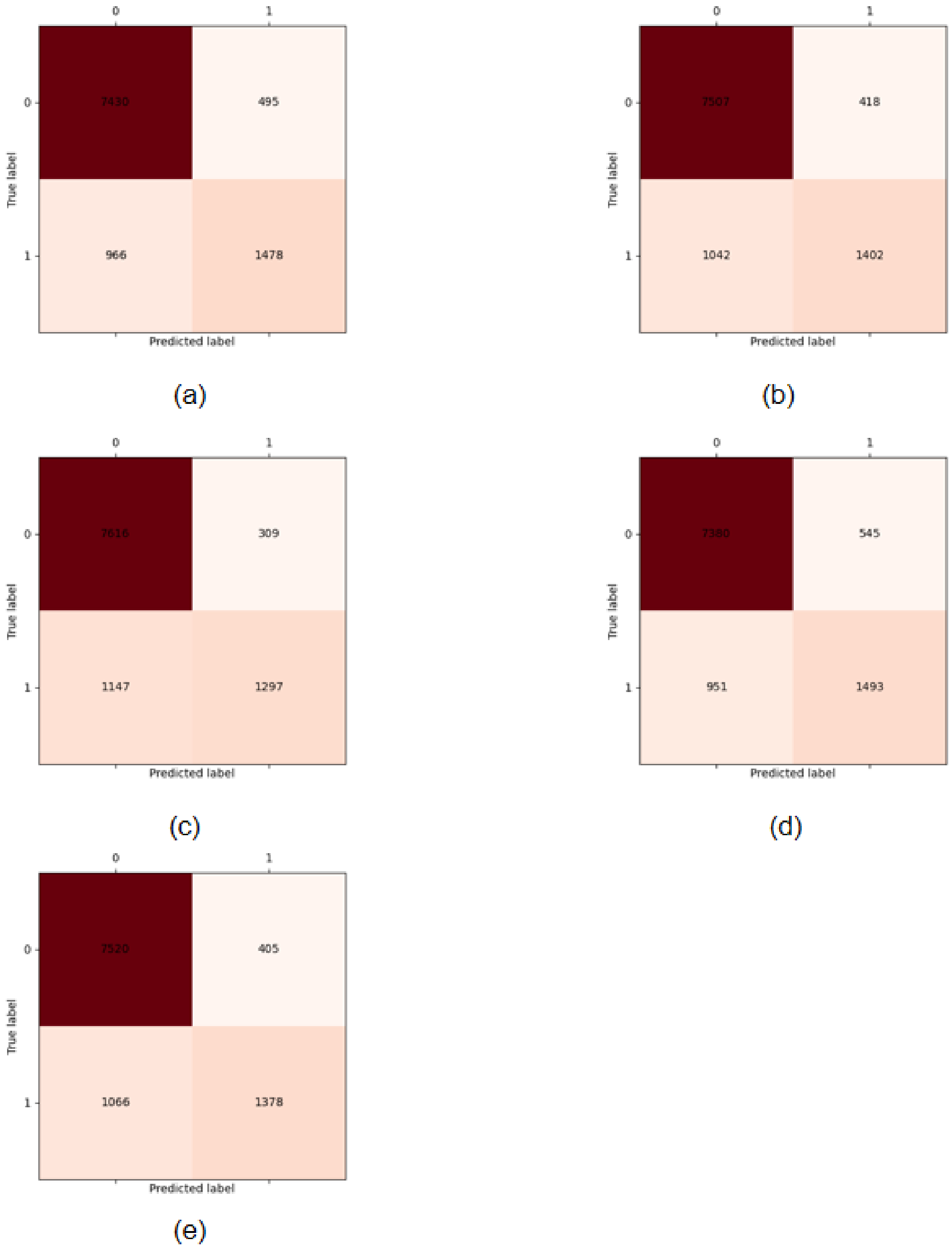
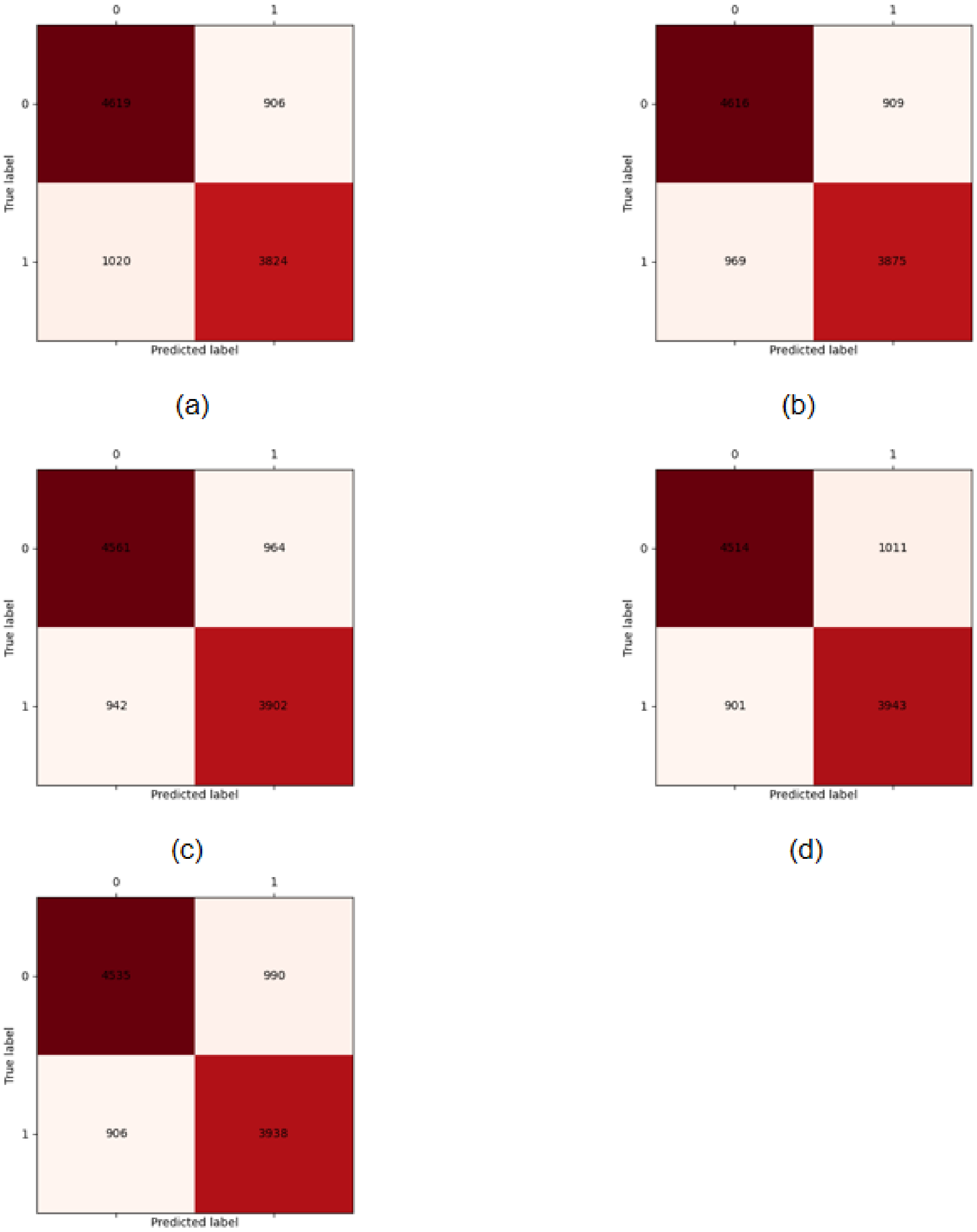


Figure B.1 Confusion matrices of the computational models for predicting plurality with 5 different random seeds, where 0 represents singular and 1 represents plural, (a) random seed is 1, (b) random seed is 29, (c) random seed is 47, (d) random seed is 65, (e) random seed is 83.



(a)

(b)

(c)

(d)

Figure B.2 Confusion matrices of the computational models for predicting definiteness with 5 different random seeds, where 0 represents indefinite and 1 represents definite, (a) random seed is 1, (b) random seed is 29, (c) random seed is 47, (d) random seed is 65, (e) random seed is 83.

C Confusion Matrices for Joint Prediction Appendix

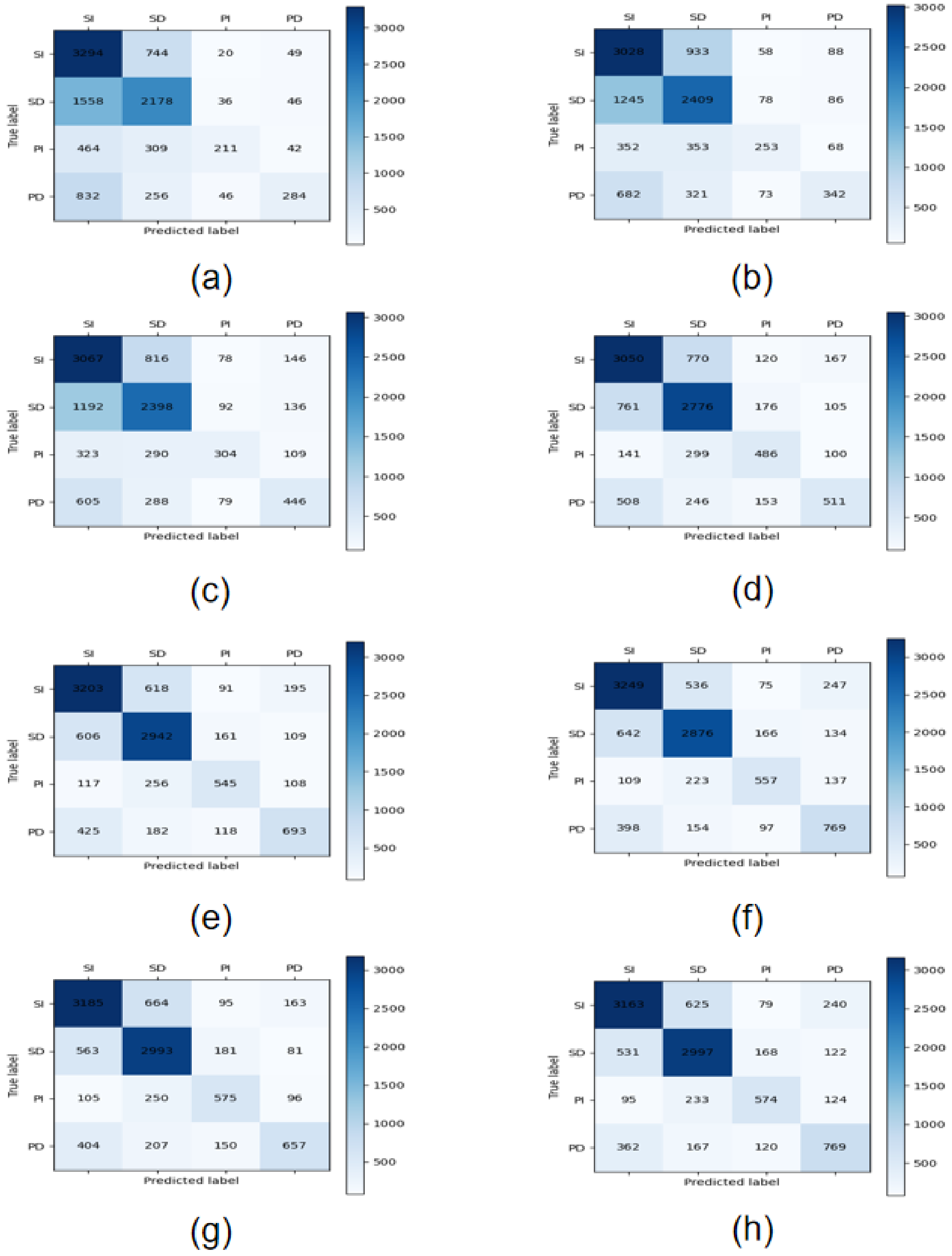


Figure C.1 The confusion matrices for 4-way prediction for all other computational models except RoBERTa (large). (a) RF, (b) LR, (c) SVM, (d) Bi-LSTM, (e) BERT, (f) BERT-wwm, (g) mBERT, (h) RoBERTa. Consistent with what is stated in Section 6, the confusion of definiteness over plurality also occurs for other models.