Using BERT for choosing classifiers in Mandarin

Jani J. Järnfors♠, **Guanyi Chen**♠, **Kees van Deemter**♠ and **Rint Sybesma**♣
♠Utrecht University ♣Leiden University
j.j.jarnfors@students.uu.nl {g.chen, c.j.vandeemter}@uu.nl
r.p.e.sybesma@hum.leidenuniv.nl

Abstract

Choosing the most suitable classifier in a linguistic context is a well-known problem in the production of Mandarin and many other languages. The present paper proposes a solution based on BERT, compares this solution to previous neural and rule-based models, and argues that the BERT model performs particularly well on those difficult cases where the classifier adds information to the text.

1 Introduction

The grammar of Mandarin and certain other Chinese languages requires that, in a number of syntactic positions, a noun must be preceded by a *classifier* word. Classifiers often give a rough indication of the kind of entity denoted by the noun. For example, the classifier "只" (zhī) in the Noun Phrase (NP) "一只狗" (yì zhī gǒu; *a dog*) indicates the noun "狗" (gǒu; *dog*) is an animal. It is worth noting that, in addition to Mandarin, classifiers also play a critical role in a few other languages, especially the East Asian languages, such as Korean, Japanese, and Vietnamese (Aikhenvald, 2000). Generally speaking, it is, in many ways, not unlike *types* in functional programming languages like Haskell, which add to each function defined by the programmer a broad semantic categorisation of that function (Thompson, 2011).

Mandarin contains a large number of classifiers, and although the choice of classifier is limited by the (head) noun with which the classifier is associated, this may still leave several options, which may sometimes produce a different meaning, e.g.,

(a) 一个 电脑/ 一台 电脑
yí gè diànnǎo / yí tái diànnǎo
'a computer'

(b) 一个 老师/ 一位 老师
yí gè lǎoshī / yí wèi lǎoshī
'a teacher'

(c) 一个 人/ 一群 人
yí gè rén / yí qún rén
'a person / people'

(d) 一杯 咖啡/ 一听 咖啡
yì bēi kāfēi / yì tīng kāfēi
'a cup/can of coffee'

Although each of these cases involves classifier choice, the problem of choosing a classifier is likely to be more challenging in those cases, such as (b)-(d), where the classifier adds information, for example, in terms of politeness ((b), neutral vs. polite), number ((c), singular vs. plural), or quantity ((d), a cup vs. a can of coffee). This is perhaps clearest in the case of (d), where "杯" (bēi; *cup*) and "听" (tīng; *can*) indicate different containers, and consequently different quantities, of coffee; these classifiers are known as measure words, as opposed to the "pure" classifiers of (a)-(c).

Researchers have asked what determines the choice of classifier, constructing algorithms that predict what classifier suits a given discourse context. The most sophisticated model we are aware of is Peinelt et al. (2017). Ambitiously, these authors decided to deal with classifiers of *all* different types, also including measure words for instance, which are difficult to predict because they add information. They approached the problem as follows: Given a sentence in which a classifier is yet to be realised, and the head noun is flagged, predict the missing classifier. For example, in the input:

(1) 一⟨CL⟩ 精彩的⟨h⟩球赛⟨/h⟩
yì ⟨CL⟩ jīngcǎi de ⟨h⟩qiúsài⟨/h⟩
'a wonderful ball game'

⟨CL⟩ indicates where the missing classifier is and the ⟨h⟩ tag pair flags the head noun. The authors construct a large-scale classifier dataset, namely

Proceedings of the 14th International Conference on Natural Language Generation (INLG), pages 172–176,
Aberdeen, Scotland, UK, 20-24 September 2021. ©2021 Association for Computational Linguistics

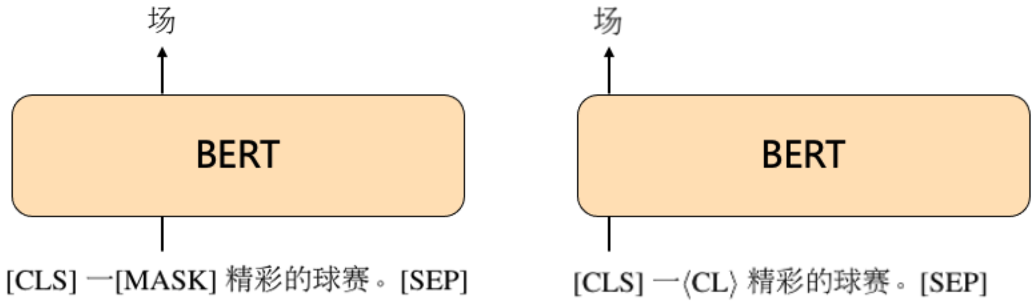
场
BERT
[CLS] 一[MASK] 精彩的球赛。[SEP]
场
BERT
[CLS] 一⟨CL⟩ 精彩的球赛。[SEP]

Figure 1: Sketch of our BERT-based Classifier selection models: predicting the classifier by unmasking the [MASK] (left); predicting the classifier as classification (right).

ChineseClassifierDataset[1] (henceforth, CCD) by extracting and filtering data from publicly available Chinese corpora. They did experiments on their CCD corpus with several baselines, including a rule-based system, two machine learning based system, and a LSTM-based system (Hochreiter and Schmidhuber, 1997). An initial evaluation study indicated that the LSTM achieved the best performance.

Our own work takes the same perspective as Peinelt et al. (2017). But although the *performance* of the model of Peinelt et al. is encouraging, it still leaves considerable room for improvement; in particular, the question comes up whether BERT, with its superior ability to take context into account, might perform better. In addition, the model of Peinelt et al. offers only limited *insight*, because it does not distinguish between different types of classifiers. In other words, the performance of the model may mask important differences between different types of classifier choice. A good way to address this limitation would be to make use of an existing categorisation of classifier types. But although linguists generally agree that "true" (or "sortal") classifiers should be distinguished from measure words (Croft, 1994; Cheng and Sybesma, 1999), there exist subtle disagreements regarding exactly how these sub-types should be defined, and what further divisions between sub-types should be taken into account. Sub-types are often described by example, without computationally implementable criteria or explicit lists of classifiers (Zhang, 2013). To our knowledge, Her and Lai (2012) are the only ones to provide comprehensive lists of classifiers of various sub-types, and in what follows we will make use of these lists.

In Section 2, we introduce two different BERT-based models, one of which uses word masking and one of which performs classification. In Section 3, we report on our comprehensive evaluation experiments, in which we compare our BERT-based models with each other and with several baselines, using the CCD dataset.

2 Choosing Classifiers using BERT

We use BERT to accomplish the task of choosing classifiers in two ways: an unsupervised way (i.e., predicting classifiers by unmasking masked tokens) and a supervised way (i.e., fine-tuning BERT on the task of classifier prediction).

2.1 Unmasking Masked Classifiers

In order to assess how well BERT, as a masked language model, can model classifiers, we tried to use BERT without any fine-tuning on the task of classifier selection. Specifically, as shown in Figure 1 (left), we replace the classifier indicator ⟨CL⟩ with the [MASK] symbol of BERT and ask BERT to unmask it. [2] The unmasked token serves as the predicted classifier. (Note that addressing the classifier selection task in this way will sometimes produce words that are not classifiers.) We refer to this model as MLM.

2.2 Classifying Classifiers

Additionally, we test BERT in its classic use. To do this, we fine-tune BERT on the CCD as a multi-class classification task, where there are 172 classes (i.e., 172 classifier words) in total, and make a prediction with the help of the [CLS] symbol (see Figure 1 (right)). We refer to this model as BERT.

[1] github.com/wuningxi/ChineseClassifierDataset

[2] Since our experiments suggested that the head flag (i.e., ⟨h⟩ and ⟨/h⟩) makes no contribution to classifier selection, we drop it to speed up the prediction.

2.3 Research Questions

At the start of our research, we formulated the following hypotheses and research questions.

1. Since BERT models context closely and is pre-trained on large scale corpora, we expect it to outperform other models;
2. How do the two BERT-based models compare? Although we expect BERT to outperform MLM, we were curious to see how well MLM performs.
3. We are curious how well BERT can handle classifiers that add information (concretely, in this paper: measure words, plurality, and politeness).

3 Experiments

3.1 Setup

Dataset. In total, there are 681,102 sentences in the CCD dataset. We split the dataset into training (60%), development (20%), and test (20%) sets following Peinelt et al. (2017).

Baselines. We tried several baseline models proposed in Peinelt et al. (2017), including: (1) a rule based model (Rule): given a head noun, assign the most frequent classifier associated with it in the training data. If two or more classifiers are equally frequent, one of the classifiers is randomly assigned. If the head noun does not appear in the training data, then the classifier "个" (gè) (which is particularly frequent and often seen as a "default" classifier) is assigned; (2) a LSTM model: for this model, we use a bi-directional LSTM (Hochreiter and Schmidhuber, 1997; Schuster and Paliwal, 1997) to encode the input; it makes predictions using the hidden representation of the last time step.

Metrics. We evaluate each model in terms of accuracy, macro-averaged precision, recall, and F1. Additionally, since the distribution of the CCD is skewed (e.g., more than 25% of the sentences use "个" (gè)), we also report the weighted averaged precision, recall, and F1.

Implementation Details. For BERT, we use the "bert-base-chinese" version[3]. When fine-tuning, we set the learning rate to 2e-5 and batch size to 150. For the LSTM, we set the batch size to 256, the hidden size to 300, and the learning rate to 2e-5. We use pre-trained Chinese word embeddings from Li et al. (2018)[4].

3.2 Results and Analysis

Table 1 charts the performance of each model. The results confirm the assumption of our first research question that BERT performs the best, defeating all models on all metrics with large margins. For example, for accuracy, compared to the second best model LSTM, BERT boosts performance from 70.44% to 81.71%. Considering its simplicity, the rule-based system achieved a considerably good performance, with higher macro-averaged precision, recall, and F1 than LSTM and with a similar accuracy as MLM. This also confirms the viability of a dictionary-based classifier selector, such as the one embedded in a previous Chinese surface realiser (Chen et al., 2018)).

MLM, as a model without any training on CCD, performs remarkably well. It receives the second best weighted average as well as micro-averaged F1 (in line with our second research question). Note that, as was mentioned, there is no guarantee that the outputs of MLM are classifiers. Concretely, during testing, MLM produces 1566 word types that are not classifiers. This is one of the reasons why its fine-tuned version, BERT, has a major improvement on the (macro-averaged and weighted averaged) recall scores as well as the accuracy. Nonetheless, it surprised us that MLM can produce a greater *variety* of classifiers than all other models. More specifically, out of 172 classifiers available in CCD, MLM has correctly produced 160 different classifiers, comparing to the 140 of Rule, 108 of LSTM, and 136 of BERT. This suggests MLM can sometimes handle rarely seen classifiers.

Regarding the last research question, we looked into measure words, plurality, and politeness respectively. First, we categorise classifiers in CCD into three sub-categories: true classifiers, measure words, and dual classifiers (i.e., classifiers that can function either as true classifiers or as measure words) based on the lists provided by Her and Lai (2012)[5]. Table 2 breaks down the performance into different sub-types of classifiers. As we can see,

[3] huggingface.co/bert-base-chinese

[4] These are word embeddings trained by skip-gram on 9 large Chinese corpora with 300 dimensions. It is available at: github.com/Embedding/Chinese-Word-Vectors

[5] These classifier lists were constructed on the basis of the Mandarin Daily Dictionary of Chinese Classifiers (MDDCC).

		Macro-averaged			Weighted-averaged		
Model	Accuracy	Precision	Recall	F1	Precision	Recall	F1
Rule	61.89	34.87	20.50	23.39	58.23	61.90	58.24
LSTM	70.44	33.11	20.12	22.48	67.90	70.44	68.12
MLM	62.22	51.91	33.40	37.68	77.28	62.23	68.21
BERT	**81.71**	**52.86**	**38.10**	**40.77**	**80.70**	**81.71**	**80.77**

Table 1: Evaluation Results of each model on CCD. The best results are **boldfaced**, whereas the second best are underlined. MLM is the model that uses BERT as a masked language model, while BERT is the fine-tuned BERT.

Category	Frequency	Accuracy
True Classifier	85,917	87.8
Dual Classifier	10,817	65.2
Measure Words	11,317	61.1

Table 2: BERT's performance on different types of classifiers; frequency of each type in the CCD test set.

although measure words appear more frequently in CCD than dual classifiers, they still receive a significantly lower accuracy.

Second, for politeness, the only frequent enough[6] politeness classifier is "位" (wèi), which expresses politeness when referring to a person. "位" appears 6737 times in the training data, but only obtains a recall score of 59.87%, which is low compared to equally frequent classifiers (classifiers with frequencies in the range of [5000, 8000) have a average recall score of 77.84%). The confusion matrix[7], shows that it is highly likely to be confused with its neutral alternative "个" (gè).

Third, regarding plurality, we pick out frequent-enough classifiers that only convey the meaning of plurality[8], including "群" (qún), "堆" (duī), "些" (xiē), "套" (tào), "对" (duì), and "双" (shuāng). Their recall scores are 52.51% (2453), 52.12% (1914), 56.51% (1910), 34.57% (1308), 62.39% (1321), and 76.49% (806), respectively, where the number in brackets is the frequency of that classifier in the training set. Meanwhile, the average recall of the range [800, 1500) and [1500, 3000) are 61.48% and 76.97%. It is interesting that BERT does a relatively good job for handling plural classifiers meaning "pair" (i.e., "对", and "双") while failing to handle plural classifiers meaning "multiple" (i.e., "群", "堆", "些", and "套"). All in all, classifiers that add information regarding measurement, plurality and politeness could not be properly selected. One explanation is that their context cannot provide enough information to pick the right classifier. Thus, for the last research question, BERT does not work well on handling classifiers that add information.

Distance between the Classifier and the Head Noun. We also explore factors that might influence the decisions of BERT. First, we consider the *distance* between the classifier and the head noun. For instance, for example (1), there is a pre-modifier consisting of two words between the classifier "场" (chǎng) and the head noun "球赛" (qiúsài; *football match*). Thus, the distance for example (1) is 2. We expect that the larger the distance is, the worse BERT performs. In our experiments, for correct predictions, the average distance (in terms of the number of words) is 1.04 while for incorrect predictions it is 1.15. An un-paired t-test confirms that distance has a negative effect on the model's performance ($p < .001$).

4 Discussion

We conclude that (1) contextualised pre-trained models (i.e., BERT and MLM) perform remarkably well on the task of choosing classifiers in Mandarin, and fine-tuning helps improve the recall of choosing classifiers; (2) a simple rule-based system has respectable performance; (3) in terms of accuracy, a pre-trained masked language model (i.e., MLM) was able to select proper classifiers about equally well as the above rule-based system; (4) BERT struggles to predict classifiers that add information (measurement, plurality, politeness).

The last finding confirms our (linguistically well-

[6] We define a classifier is *frequent enough* if it appears more than 50 times in the training set.

[7] The full confusion matrix is too large to print here but, together with the system outputs, is available at: github.com/a-quei/bert-chinese-classifier

[8] Some classifiers have multiple meanings, one of which expresses plurality.

established) expectation that some classifier occurrences cannot be predicted from their linguistic context alone since they themselves carry additional information. Since the choice of classifier is not deterministic (e.g., consider the choice between "个" and "台" in example (a)), the type of corpus evaluation that was performed in this paper arguably does not "tell the whole story" regarding the quality of the different models. To remedy this issue, we plan two further experiments, each of which starts from the observation that the classifier that was chosen in a given linguistic context in the corpus will often not be the only felicitous choice.

One experiment will focus on *speakers*. We will ask several participants to choose classifiers given a linguistic context. By comparing the outcomes of such an elicitation experiment with the CCD corpus, we will obtain a better understanding of the variations that exist between speakers and of the difficulty of the task that we have set our algorithms. By thus asking multiple participants to accomplish the same task as our algorithms, we will obtain a new corpus, in which each linguistic context is associated with a bag of (1 or more) possible classifiers. This new dataset will enable us to conduct a new, non-deterministic evaluation of the models.

Another additional experiment will have human *readers* judge the acceptability of each classifier choice that is made by a given model. Reader experiments of this kind are a standard tool in judging the quality of decisions taken by a natural language generation algorithm (e.g. van der Lee et al. (2019)) and will give rise to a new set of analyses analogous to the ones in the present paper, which will give us a better understanding of the quality of the decisions that are taken by each model.

In the future, we also plan to extend the models we tested in this study. For example, regarding the pre-trained language model, a promising candidate to investigate is ERNIE (Sun et al., 2020), which has proved to be more powerful in modelling Mandarin Chinese. Regarding the unsupervised MLM setting, the following option would be worth trying: instead of choosing the most probable word type from the whole vocabulary, one could ask the model to output the most probable classifier from all classifiers.

References

Alexandra Y. Aikhenvald. 2000. Classifiers. a typology of noun categorization devices. *NewYork: Oxford University Press*.

Guanyi Chen, Kees van Deemter, and Chenghua Lin. 2018. SimpleNLG-ZH: a linguistic realisation engine for Mandarin. In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 57–66, Tilburg University, The Netherlands. Association for Computational Linguistics.

Lisa Lai-Shen Cheng and Rint Sybesma. 1999. Bare and not-so-bare nouns and the structure of np. *Linguistic inquiry*, 30(4):509–542.

William Croft. 1994. Semantic universals in classifier systems. *Word*, 45(2):145–171.

One-Soon Her and Wan-Jun Lai. 2012. Classifiers: The many ways to profile'one'—a case study of taiwan mandarin. *International Journal of Computer Processing Of Languages*, 24(01):79–94.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Chris van der Lee, Albert Gatt, Emiel van Miltenburg, Sander Wubben, and Emiel Krahmer. 2019. Best practices for the human evaluation of automatically generated text. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 355–368, Tokyo, Japan. Association for Computational Linguistics.

Shen Li, Zhe Zhao, Renfen Hu, Wensi Li, Tao Liu, and Xiaoyong Du. 2018. Analogical reasoning on Chinese morphological and semantic relations. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 138–143, Melbourne, Australia. Association for Computational Linguistics.

Nicole Peinelt, Maria Liakata, and Shu-Kai Hsieh. 2017. ClassifierGuesser: A context-based classifier prediction system for Chinese language learners. In *Proceedings of the IJCNLP 2017, System Demonstrations*, pages 41–44, Tapei, Taiwan. Association for Computational Linguistics.

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

Yu Sun, Shuohuan Wang, Yukun Li, Shikun Feng, Hao Tian, Hua Wu, and Haifeng Wang. 2020. ERNIE 2.0: A continual pre-training framework for language understanding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8968–8975.

Simon Thompson. 2011. *Haskell: the craft of functional programming*, third edition. Addison-Wesley. More information at www.haskellcraft.com.

Niina Ning Zhang. 2013. *Classifier Structures in Mandarin Chinese*, volume 263. Walter de Gruyter.