Computer Speech and Language 79 (2023) 101466

Contents lists available at ScienceDirect

Computer Speech & Language

journal homepage: www.elsevier.com/locate/csl

Neural referential form selection: Generalisability and interpretability

Guanyi Chen [a,*], Fahime Same [b], Kees van Deemter [a]

[a] Department of Information and Computing Sciences, Utrecht, The Netherlands
[b] Department of Linguistics, University of Cologne, Cologne, Germany

ARTICLE INFO

Keywords:
Natural Language Generation
Referring Expression Generation
Deep learning
Probing classifier
Multilinguality

ABSTRACT

In recent years, a range of Neural Referring Expression Generation (REG) systems have been built and they have often achieved encouraging results. However, these models are often thought to lack transparency and generality. Firstly, it is hard to understand what these neural REG models can learn and to compare their performance with existing linguistic theories. Secondly, it is unclear whether they can generalise to data in different text genres and different languages. To answer these questions, we propose to focus on a sub-task of REG: Referential Form Selection (RFS). We introduce the task of RFS and a series of neural RFS models built on state-of-the-art neural REG models. To address the issue of interpretability, we probe these RFS models using probing classifiers that consider information known to impact the human choice of Referential Forms. To address the issue of generalisability, we assess the performance of RFS models on multiple datasets in multiple genres and two different languages, namely, English and Chinese.

1. Introduction

Referring Expression Generation (REG) is one of the main stages of the classic Natural Language Generation (NLG) pipeline (Reiter and Dale, 2000; Krahmer and van Deemter, 2012; van Deemter, 2016).[1] REG is not only of practical value for practical NLG (Reiter, 2017) (including applications with computer vision (Mao et al., 2016), and robotics (Fang et al., 2015) for example), but can also be used as a tool for expressing and testing theories of human language use (van Deemter, 2016).

REG includes one-shot REG and REG-in-context. The one-shot REG task involves finding a set of attributes to single out a referent from a set of distractors. This task seeks to identify a referent in a single shot, disregarding its linguistic context (Krahmer and van Deemter, 2012). Unlike one-shot REG, REG-in-context takes the linguistic context of an expression into consideration: REG-in-context involves generating appropriate referring expressions (REs) to refer to a referent at different points in the discourse (Belz and Varges, 2007). In this work, we focus on the latter task, REG-in-context, with the aim of addressing some key problems of current REG-in-context models, namely their lack of interpretability and generalisability.

Classic REG-in-context is usually a two-step procedure. First, the form of the RE (i.e., the referential form, henceforth, RF) is determined, which is also called the task of Referential Form Selection (RFS). For instance, when referring to Joe Biden at a given point in the discourse, the first step is to decide whether to use a proper name ("*Joe Biden*"), a description ("*the president of the USA*"), a demonstrative ("*this person*") or a pronoun ("*he*"). The second step is to determine the content of the RE, choosing between different

* Corresponding author.
E-mail addresses: g.chen@uu.nl (G. Chen), f.same@uni-koeln.de (F. Same), c.j.vandeemter@uu.nl (K. van Deemter).

[1] The complete NLG pipeline is often thought to contain six stages: document planning, document structuring, lexical choice, aggregation, REG, and linguistic realisation.

https://doi.org/10.1016/j.csl.2022.101466
Received 15 April 2022; Received in revised form 4 October 2022; Accepted 17 October 2022
Available online 17 November 2022

ways in which a given form can be realised. For instance, to generate a description of Joe Biden, one needs to decide whether to mention only his job (e.g., *The president* entered the Oval Office.), or to mention the country as well (e.g., *The president of the United States* arrived in Cornwall for the G7 Summit.)

In early studies, computational linguists have often linked REG to linguistic theories. For example, Henschel et al. (2000) investigated the impact of three linguistic features namely recency, subjecthood, and discourse status on pronominalisation. The same holds for feature-based models (Belz et al. (2010)) where models are trained on linguistically encoded data.

In the last 10 years, neural networks have become popular for tackling Natural Language Processing (NLP) tasks, and REG is no exception. A number of neural network-based REG models have been proposed (Castro Ferreira et al., 2018a; Cao and Cheung, 2019; Cunha et al., 2020; Same et al., 2022), generating REs in an End2End manner without any feature engineering. These models tend to follow the sequence-to-sequence framework (Sutskever et al., 2014), where there is an encoder for encoding the given discourse, and a decoder responsible for generating REs using the encoded information. Models were assessed on a benchmark dataset called WEBNLG. Evaluation results suggested that these neural methods perform well. However, we argue that previous work has three major shortcomings. These concern, briefly, the relative opacity of many of the models used, the choice of corpora on which the models are based, and the focus on a small number of languages.

1. Lack of Interpretability. As is the case with many uses of neural networks in NLP, the opacity of these models can make it hard to understand any shortcomings of the models and to improve them further. Moreover, from a theoretical perspective, where the goal is to develop models of human behaviour (Vicente and Wang, 1998; Sun, 2008), neural models are only of interest insofar as they are explainable. After all, reference has been studied for a long time and from many different theoretical angles (Ariel, 1990; Gundel et al., 1993; Brennan, 1995; Arnold and Griffin, 2007; Fukumura and van Gompel, 2011; Kibrik et al., 2016; von Heusinger and Schumacher, 2019; Same and van Deemter, 2020), but to the extent that NeuralREG models are opaque (i.e., not explainable), it is difficult to compare these models with the findings and insights from these earlier studies. Consequently, models are unable to benefit from earlier insights and, conversely, earlier theories are unable to benefit from any successes achieved by neural models.

2. Choice of Corpora. All previous work was tested on a benchmark dataset called WEBNLG (Gardent et al., 2017; Castro Ferreira et al., 2018b). This dataset was originally built for generating text from RDF triples (namely, *RDF-to-Text Generation*). Its data was extracted from DBpedia[2] and its texts are mostly formal descriptions of a set of triples. Therefore, WEBNLG may not reflect the everyday use of REs. For example, 85% of its REs are first-mentions, and 71% of them are proper names (see Section 4.2). We believe that this makes WEBNLG an unfortunate choice of corpus for assessing REG-in-context systems.

3. Exclusive Focus on Western European Languages. Speakers of different languages adopt different referring mechanisms (Walker et al., 1994; Prasad, 2003). The difference between speakers of East Asian languages (e.g. Chinese and Japanese) and speakers of Western European languages (e.g. English and Dutch; Newnham (1971)) has attracted particular attention. For example, theoretical linguists (Huang, 1984) have argued that East Asian languages rely more heavily on context than Western European languages, and, as a result, speakers of East Asian languages frequently use Zero Pronouns (ZPs), i.e. REs that contain no words and are resolved solely based on their context (see Chen and van Deemter (2020), Chen et al. (2018), Chen and van Deemter (2022), Chen (2022) for empirical testing and computational modelling). Consider the following question in Chinese: "你看见比尔了吗?" (nǐ kànjiàn bǐěr le mā; *Have you seen Bill?*). A Chinese speaker can reply '∅看见∅了。" (Ø kànjiàn Ø le; Ø *saw* Ø.) where the two Ø are ZPs referring to the speaker himself/herself and "Bill" respectively. Therefore, the question of whether the state-of-the-art (SOTA) NeuralREG models work on other languages, especially East Asian languages, is still unanswered.

To address the first issue (lack of interpretability), we introduce a series of probing tasks. As a probing task, a diagnostic classifier is trained on representations from the model and its performance tells us how well these representations encode the information associated with the task. Probing is a well-established method for analysing whether the latent representations of a model encode specific information. This approach has been widely used for analysing models in machine translation (Belinkov et al., 2017), language modelling (Giulianelli et al., 2018), relation extraction (Alt et al., 2020), and so on. Probing experiments have also been used to a lesser extent to analyse models of coreference resolution (Sorodoc et al., 2020), showing that language models capture morphosyntactic information and, to some extent, semantico-referential aspects of anaphora.

Our focus is on the encoding of linguistic features by neural RE models, and since most reference production studies in the linguistic tradition focus on the task of RF choice, we will address the same task, referred to here as RFS. The RFS task is defined as follows: given a text whose REs have not yet been generated, and given the intended referent for each of these REs, the RFS task is to develop an algorithm that finds the proper RF from a set of K candidate RFs. RFS is a classification problem, i.e., the algorithm's task is to select a referential class from a set of given classes. For example, in the case of a pronominalisation task, there are two classes, pronominal and non-pronominal forms (K=2), and the RFS task is to decide which form to use.

To tackle RFS, we adapt the SOTA NeuralREG models of Castro Ferreira et al. (2018a). We propose a strong baseline that uses only a single encoder (while Castro Ferreira et al. (2018a) used multiple encoders). Additionally, we leverage pre-trained word embeddings (e.g., GloVe) and language models (e.g., BERT).

Regarding the second issue (i.e., the choice of corpora), we first assess and probe RFS models on the WEBNLG corpus. We find that many experimental results are not in line with what linguistic theories suggest. We study this issue further in combination with the third issue (exclusive focus on Western European languages) by building a realistic multilingual dataset from the ONTONOTES corpus consisting of REs from various genres and in both English and Chinese. We evaluate and probe RFS models on this new dataset and compare the results across different corpora (i.e., WEBNLG and ONTONOTES) and different languages (i.e., English and Chinese).

[2] https://www.dbpedia.org/

Table 1
An example data from the WEBNLG corpus. In the delexicalised text, every entity is underlined.

Triples: (AWH_Engineering_College, country, India) (Kerala, leaderName, Kochi) (AWH_Engineering_College, academicStaffSize, 250) (AWH_Engineering_College, state, Kerala) (AWH_Engineering_College, city, "Kuttikkattoor") (India, river, Ganges)
Text: AWH Engineering College is in Kuttikkattoor, India in the state of Kerala. The school has 250 employees and Kerala is ruled by Kochi. The Ganges River is also found in India.
Delexicalised Text: **Pre-context**: AWH_Engineering_College is in "Kuttikkattoor" , India in the state of Kerala . **Target Entity**: AWH_Engineering_College **Post-context**: has 250 employees and Kerala is ruled by Kochi . The Ganges River is also found in India .

In Section 2, we summarise the background of our study. We then list our research questions and expectations in Section 3. In Section 4, we formally define the task of RFS and describe our datasets (WEBNLG and ONTONOTES). In Section 5, we describe how we adapted NeuralREG models to the RFS task and we report on their performance. In Section 6, we introduce our probing tasks and probing classifiers, and report the probing results of each RFS model on two datasets. In Section 7, we compare the results across different corpora and different languages.

2. Background

2.1. REG-in-context

Given a text whose REs have not yet been generated, and given the intended referent for each of these REs, the REG-in-context task is to build an algorithm that generates all these REs (Belz and Varges, 2007). This task has attracted many research efforts; for instance, the GREC shared tasks (Belz et al., 2010) sparked a plethora of feature-based solutions for the task (Hendrickx et al., 2008; Greenbacker and McCoy, 2009). More recently, this task has been formulated into a format that goes together well with deep learning. Castro Ferreira et al. (2018a) introduced the End2End REG task, built a corresponding dataset based on WEBNLG (Castro Ferreira et al., 2018b), and constructed NeuralREG models.

The WEBNLG corpus was originally designed to assess the performance of NLG systems (Gardent et al., 2017). Each sample in this corpus corresponds to an item in a knowledge base described by a Resource Description Framework (RDF) triple (Table 1). Castro Ferreira et al. (2018a) and Castro Ferreira et al. (2018b) enriched and delexicalised the corpus to fit the REG-in-context task.

Table 1 shows a text created from an RDF, and its corresponding delexicalised version. Taking the delexicalised text in this table as an example, given the entity *AWH_Engineering_College*, the REG-in-context task chooses an RE based on that entity and its pre-context ("*AWH_Engineering_College is in "Kuttikkattoor", India in the state of Kerala.*") and its post-context ("*has 250 employees and Kerala is ruled by Kochi. The Ganges River is also found in India.*").

2.2. Linguistic factors that impact the choice of RFs

Languages display a large inventory of expressions for referring to entities (von Heusinger and Schumacher, 2019). In linguistics, a speaker's realisation choice is associated with the prominence (i.e., activation of mental representations in the listener's mind) of a referent at a particular point in the discourse: attenuated forms such as pronouns are often used to refer to highly prominent referents, while richer forms such as descriptions and proper names are used to refer to less prominent ones (Ariel, 1990; Gundel et al., 1993; von Heusinger and Schumacher, 2019). A large body of research has tried to assess the influence of different features modulating the prominence of a referent. In the following, we only talk about factors we use in our probing experiments, and do not discuss factors such as animacy (Fukumura and van Gompel, 2011), competition (Arnold, 2010), and coherence relations (Kehler et al., 2008).

Referential status or *givenness* has been widely discussed in the literature (see Chafe (1976), Prince (1981)). When a new character is introduced into the discourse, the chance that this happens by means of a pronoun is very low. Pronouns are reserved for referring to previously introduced (i.e., "given") referents. *Recency*, another well-studied cue, is defined as the distance between the target referent and its coreferring antecedent. If a referent is not too far from its antecedent, reduced forms are typically used to refer to it. There are also intra-clausal cues such as *grammatical role* (Brennan, 1995) and *thematic role* (Arnold, 2001) which impact the prominence status of referents. For instance, the subject of a sentence is more prominent than the object. Discourse-structural features affect the organisational aspects of discourse, which in turn can influence the prominence of referents. Centering-based theories (Grosz et al., 1995) often use the notion of local focus to explain pronominalisation. *Local focus* takes the current and previous utterances into account. *Global focus*, on the other hand, situates a referent within a larger discourse segment (Hinterwimmer, 2019).

2.3. Interpretability

Unlike classic approaches such as rule-based and feature-based methods, SOTA neural E2E models arguably lack transparency at two levels: (1) they do not explicitly use linguistic features such as those mentioned above, and (2) we do not have easy access to their decision steps. The success of these models comes from a combination of efficient learning algorithms and their huge parametric space (Castelvecchi, 2016; West, 2018; Barredo Arrieta et al., 2020). Deep learning models are evolving rapidly and their learning, reasoning, and adaptation capabilities are constantly improving. With the growth in popularity of these models, interpretability and explainability become of paramount importance.

In recent years, numerous research efforts in the field of eXplainable AI (XAI) have sought to make neural systems more interpretable and explainable (see Dosilovic et al. (2018) and Barredo Arrieta et al. (2020) for an overview). It is generally believed that the vector representations of neural models encode some "continuous analogue of linguistic structures" (Torroba Hennigen et al., 2020). Therefore, most post-analyses of these models, such as probing experiments, try to answer the question of what kind of linguistic features are encoded in neural networks.

For probing experiments in the context of this paper, we first train neural RFS models and generate representations. We then define several probing classifiers based on the features that are assumed to influence the choice of RF (e.g., the ones explained in Section 2.2). These classifiers take the representations as input and classify them in accordance with the probing task. If they perform well, we can conclude that the model has learned the information relevant to the classifiers' task. Torroba Hennigen et al. (2020) consider this type of probing as an *extrinsic* evaluation method. This method thus provides a way of looking at the latent representations of a neural model through the lens of linguistic features.

3. Research questions

In this study, we consider the following research questions ($\mathcal{RQ}$s).

3.1. $\mathcal{RQ}_1$: The choice of model architectures

Analogous to much other research in NLP, we investigate various model architectures for RFS. We test several models to answer the following three questions: (1) *Do models for REG work well in selecting RFs?* we adapt the SOTA NeuralREG models in Castro Ferreira et al. (2018a) to model RFS; (2) *Do simpler model architectures work worse than SOTA models for RFS?* we propose a simple RFS model that uses only a single GRU (Cho et al., 2014) encoder and compare its performance to SOTAs (see Section 5 for more details); (3) *Do pre-trained word embeddings or language models help RFS?* although there are several works applying neural models to REG, only Cao and Cheung (2019) have used pre-trained embeddings, but no ablation study has been conducted. In this study, we test models using pre-trained word embeddings (`GloVe` and `SGNS`) and language models (`BERT`).

3.2. $\mathcal{RQ}_2$: The use of structurally different corpora

We are interested in *how the choice of corpus affects the performance of RFS models and the information learned by each RFS model.* We first perform RFS on the WEBNLG corpus (Gardent et al., 2017; Castro Ferreira et al., 2018b). As discussed in Section 1, WEBNLG has certain shortcomings. For example, it is formal and monolingual, and its texts are extremely short. We, therefore, built a new REG/RFS dataset from the ONTONOTES corpus, whose texts are thought to be more representative of normal language use than WEBNLG, and which is multi-lingual, and multi-genre. We assess and probe our RFS models on both datasets and compare their results.

3.3. $\mathcal{RQ}_3$: Handling unseen entities

Castro Ferreira et al. (2018a) defined the REG task representing each entity with an entity label (e.g., AWH_Engineering_College in Table 1). It has already been pointed out that this makes it difficult for REG models to handle unseen entities (Cao and Cheung, 2019; Cunha et al., 2020) because entity labels of unseen entities are usually out-of-vocabulary (OOV) words for REG models. The same is likely to happen when modelling RFS. In addition, as mentioned above, we plan to examine pre-trained word embeddings and language models. Using entity labels prevents entity representations from benefiting from these pre-trained models (again, since the entity labels of unseen entities are usually OOV words).

To ease the handling of unseen entities, Cunha et al. (2020) replaced entity labels of the target referents with proper names by simply substituting underscores in entity labels with white spaces (e.g., changing "AWH_Engineering_College" to "AWH Engineering College").[3] In this study, to gain further benefits from pre-trained models, we use proper names not only for the target referents but also for the referents in the pre-context and post-context. We are interested to test whether using the lexical format instead of entity labels helps RFS models to better handle unseen entities.

[3] The term *proper* in this context refers to the non-underscored version of entity labels, which can also be used as referring expressions. This term is not to be confused with the class proper names used in the RFS classification tasks.

Table 2
Three different types of English RF classification.

Type	Classes
4-Way	Demonstrative, Description, Proper Name, Pronoun
3-Way	Description, Proper Name, Pronoun
2-Way	Non-pronominal, Pronominal

3.4. $\mathcal{RQ}_4$: The use of REs in different languages

As discussed in Section 1, speakers of different languages use REs differently and, therefore, we are curious *how language impacts the behaviour of each RFS model (i.e., its performance and the linguistic information it learns)*. Concretely, in addition to English, we test RFS models on Chinese in order to validate their ability to model Zero Pronouns (ZPs), which Chinese speakers use much more frequently than English speakers.

3.5. $\mathcal{RQ}_5$: Linguistic information

As mentioned in Section 1, we want to investigate *what linguistic information neural RFS models learn*. We explore a number of features that have been shown to influence the choice of RFs in linguistics (see Section 2.2), and formalise a series of probing tasks accordingly (see Section 6.2). Building on the previous two research questions, we would also like to know how the following three factors influence the models' ability to acquire a given linguistic feature: (1) the model architecture, (2) the use of pre-trained word embeddings and language models, and (3) the classification task (e.g., full RFS or pronominalisation).

3.6. Expectations

In connection with these research questions, we have a number of more detailed expectations, which we number here for convenience. In the bracket after each expectation number, we indicate which research question the expectation relates to.

- $\mathcal{E}_1$ ($\mathcal{RQ}_1$): In this study, we try both an attention-based RFS model, the SOTA in REG, and a simple model with a single GRU (see Section 3.1). We expect that the *attention-based models outperform the simpler GRU-based models*.
- $\mathcal{E}_2$ ($\mathcal{RQ}_1$): Since pre-trained word embeddings have been shown to be effective in many NLP tasks, and contextual pre-trained language models (e.g., BERT) have been proved to be able to further boost performance, we expect that *models with pre-trained embeddings work better than those without pre-training, but worse than those with BERT*.
- $\mathcal{E}_3$ ($\mathcal{RQ}_2$): Since ONTONOTES is considered to be more realistic and contains more complex uses of REs, we expect *models trained on* ONTONOTES *to have lower performance than models trained on* WEBNLG.
- $\mathcal{E}_4$ ($\mathcal{RQ}_3$): We expect that *representing entities with their proper names enables models to handle unseen entities better than representing entities with their entity labels. Since* ONTONOTES *is a dataset that mixes seen and unseen entities, this expectation implies that models trained using proper names perform better than models trained using entity labels*.
- $\mathcal{E}_5$ ($\mathcal{RQ}_4$): Given the theory that Chinese speakers process ZPs in the same way as pronouns (Yang et al., 1999), we expect that *RFS models that work well in English would also work well in Chinese*.
- $\mathcal{E}_6$ ($\mathcal{RQ}_4$): Since Chinese relies more on context than English (see Section 3.4), we expect that *Chinese RFS models would benefit more from the use of contextual representations (i.e., BERT) than English RFS models*.
- $\mathcal{E}_7$ ($\mathcal{RQ}_5$): By conducting a probing analysis (see Section 3.5), we expect that *models with better performance would also learn more relevant linguistic information*.
- $\mathcal{E}_8$ ($\mathcal{RQ}_5$): We try different classification tasks. Since more fine-grained classification provides more detailed supervision signals when training RFS models, it is plausible to expect that *RFS models learn more useful linguistic information when trained for more fine-grained classifications*.
- $\mathcal{E}_9$ ($\mathcal{RQ}_5$): The probing tasks target the factors that supposedly work similarly across different languages. Therefore, we expect to see *the same patterns in the probing results across both languages*.

4. RFS: Task and datasets

In this section, we introduce the task of RFS and describe the WEBNLG dataset and how we built the ONTONOTES dataset.

4.1. The RFS task

Following Castro Ferreira et al. (2018a), we define the RFS task as follows: given the previous context $x^{(pre)} = \{w_1, w_2, \dots, w_{i-1}\}$, where each w is either a word or a delexicalised entity label, the target referent $w^{(r)} = \{w_i\}$, and the post context $w^{(pos)} = \{w_{i+1}, w_{i+2}, \dots, w_n\}$, an RFS algorithm aims at finding the proper RF $\hat{f}$ from a set of K candidate RFs $\mathcal{F} = \{f_k\}_{k=1}^{K}$.

Table 3
Four different types of Mandarin RF classification.

Type	Classes
5-way	Demonstrative, Description, Proper Name, Pronoun, Zero Pronoun
4-Way	Description, Proper Name, Pronoun, Zero Pronoun
3-Way	Proper Name, Pronoun, Zero Pronoun
2-Way	Overt Referring Expression, Zero Pronoun

The above definition, hereafter referred to as RFS-EL, uses delexicalised entity labels to represent referents in the input. As discussed in Section 3.3, such a method makes it hard for the models to handle unseen entities during inference. Therefore, we also try another setting where the entity labels are replaced with their corresponding proper names. From now on, this task is called RFS-PN.

In contrast to RFS-EL, RFS-PN takes word-based input, including the previous context $x^{(pre)} = \{w_1, w_2, \dots, w_{i-1}\}$, where each w is a word, the target referent $x^{(r)} = \{w_i, w_{i+1}, \dots, w_j\}$, and the post context $x^{(pos)} = \{w_{j+1}, w_{j+2}, \dots, w_n\}$. In Sections 4.2 and 4.3, we describe the datasets used for RFS-EL. For RFS-PN, we simply replace all entity labels with proper names in each dataset.

Regarding possible RFs for the RFS task, we test various classifications. For English, we test three different classifications, depicted in Table 2. Since demonstrative noun phrases are infrequent, we decided to also conduct a 3-way classification, merging descriptions and demonstratives. Also, most emphasis in the linguistic literature is on the pronominalisation issue. Therefore, we also include a 2-way classification task in the study. For Chinese, we consider an extra RF: ZP. This results in four different classifications which are listed in Table 3.

4.2. The WEBNLG dataset

We use the v1.5 of WEBNLG (Castro Ferreira et al., 2018b), in which the RF of each RE is provided. Castro Ferreira et al. (2018b) divided documents in the test set into *seen* (where all data are from the same domains as the training data) and *unseen* (where all data are from different domains than the training data).

We split the dataset in the same way as in Castro Ferreira et al. (2018b), but we have decided not to use the unseen data from WEBNLG. First, the way the test set of WEBNLG was constructed results in almost all referents from the seen test set appearing in the training set (9580 out of 9644), while only a few referents from the unseen test set appear in the training set (688 out of 9644). A document in which almost all referents are unseen is not realistic, or at least not the focus of this study. Second, the size of the underlying triples of the seen and unseen test sets differs from each other. The seen data is built from triple group sized in the range 2–7, whereas the unseen data is built from triples in the range of 2–5. In other words, seen and unseen data do not have the same complexity. So when we test RFS models on them, the results for the two subsets are not comparable. After excluding the unseen data, the resulting WEBNLG corpus contains 67,027, 8278, and 9644 samples in the training, development, and test sets, respectively.[4]

Limitations of WEBNLG. As mentioned in Section 1, WEBNLG has some notable shortcomings. First, it consists of rather formal texts that may not reflect the everyday use of REs, and in which very simple syntactic structures dominate. Second, the texts in WEBNLG are extremely short, with an average length of only 1.4 sentences. Third, as many as 85% of the REs in WEBNLG are first mentions, while 71% of the REs are proper names.

4.3. The ONTONOTES dataset

To construct a realistic multilingual REG/RFS dataset, we used the Chinese and English portions of the ONTONOTES dataset[5] whose contents come from six sources, namely broadcast news, newswires, broadcast conversations, telephone conversations, web blogs and magazines. We have called the resulting Chinese subset ONTONOTES-ZH and the English subset ONTONOTES-EN. In what follows, we describe the construction process.

First, for each RE in ONTONOTES, we used three previous sentences as the pre-context and three subsequent sentences as the post-context. Using the constituency syntax tree of the sentence containing the target referent, the POS tags, and the surface form of the target referent, we automatically annotated each RE with its RF category. Note that information such as morpho-syntactic annotation (lemma, and POS tags), constituency syntax trees, and coreference annotation is available in ONTONOTES. Therefore, the quality of the RF category annotation depends mainly on the quality of these annotations.

Second, we excluded all coreference chains consisting only of pronouns and ZPs. The pronominal chains consist mainly of first/second-person referents, and we do not expect much variation in RFs of these cases. In other words, we only use the chains that have *at least* one overt non-pronominal RE.

Third, we delexicalised the corpus following Castro Ferreira et al. (2018a).[6] Additionally, since we use Chinese BERT as one of our RFS models and it only accepts input shorter than 512 Chinese characters, we create two versions of ONTONOTES-ZH: ONTONOTES-ZH,

[4] To answer research question $\mathcal{RQ}_3$ in Section 3.3 which focuses on unseen entities, we only use the data from ONTONOTES.

[5] It is licensed under the Linguistic Data Consortium: https://catalog.ldc.upenn.edu/LDC2013T19

[6] We have not yet carried out a thorough manual verification of the delexicalisation quality. However, we believe that manual verification can improve the quality of the delexicalisations and thus the performance of the models.

Table 4
An example data from the ONTONOTES-EN corpus. In the delexicalised text, every entity is underlined.

Text: Citizens & Southern Corp. said it signed a definitive agreement to acquire Security Pacific Corp.'s New York-based factoring unit. Terms of the bank holding companies' agreement were not disclosed. Factoring involves the purchase and collection of another company's receivables. Citizens, based in Atlanta, said it has about $ 4.6 billion in factored sales annually; the Security Pacific unit has about $ 1.8 billion annually. Security Pacific 's factoring business works with companies in the apparel, textile and food industries, among others.
Delexicialised Text: **Pre-context**: Citizens_Southern_Corp said Citizens_Southern_Corp signed the_bank_holding_companies_agreement. Terms of the_bank_holding_companies_agreement were not disclosed. Factoring involves the purchase and collection of another company 's receivables. **Target Entity**: Citizens_Southern_Corp **Post-context**: said Citizens_Southern_Corp has about $ 4.6 billion in factored sales annually; the_Security_Pacific_unit has about $ 1.8 billion annually. the_Security_Pacific_unit works with companies in the apparel, textile and food industries, among others.

Table 5
Statistics of WEBNLG and ONTONOTES. O-EN and O-ZH stand for ONTONOTES-EN and ONTONOTES-ZH.

	WEBNLG	O-EN	O-ZH
Percentage of first mentions	85%	43%	43%
Percentage of proper names	71%	21%	15%
Average number of tokens	18.62	106.44	139.55

which is the original Chinese portion of ONTONOTES, and ONTONOTES-ZH$_{\leq 512}$, where we remove all samples whose total length is longer than 512 characters. The length is calculated by removing all underscores introduced during delexicalisation and summing the length of the pre-contexts, post-contexts, and target referents.

Last, we split the whole dataset into a training set and a test set in accordance with the CoNLL 2012 Shared Task (Pradhan et al., 2012). We then sampled 10% of the documents from the training set as the development data. As a result, we obtained ONTONOTES-EN where the training, development and test sets contain 71667, 8149, and 7619 samples, respectively, and ONTONOTES-ZH where the training, development and test sets contain 70428, 9217, and 11607 samples, respectively. 38.44% and 41.45% of the referents in the test sets of ONTONOTES-EN and ONTONOTES-ZH also appear in the training sets. Table 4 shows a sample from the ONTONOTES-EN dataset.

4.4. WEBNLG *vs.* ONTONOTES*: an initial comparison*

Building on the nature of ONTONOTES and the statistics in Table 5, we observe that: (1) the WEBNLG data is from DBPedia, while the ONTONOTES data is multi-genre; (2) ONTONOTES has a much smaller proportion of first mentions and proper names; and (3) the documents in ONTONOTES are on average much longer than the documents in WEBNLG. Having said this, ONTONOTES largely mitigates the problems of WEBNLG discussed in Section 4.2.

5. Introducing and testing neural RFS models

In this section, we start by introducing RFS-EL and RFS-PN models. We build Neural RFS models by (1) adopting the best NeuralREG model from Castro Ferreira et al. (2018a), and (2) proposing a new alternative that is simpler and can more easily incorporate pre-trained representations. We then describe the baselines and metrics, and test the RFS models on the two datasets.

5.1. RFS-EL models

ConATT-EL. We adopt the CATT model of Castro Ferreira et al. (2018a), which achieves the best performance on REG among the models tested in their study. Given the input, we first use a Bidirectional GRU (BiGRU, Cho et al., 2014) to encode $x^{(pre)}$ and $x^{(pos)}$. Formally, for each $k \in [pre, pos]$, we encode $x^{(k)}$ to $h^{(k)}$ with a BiGRU:

$$h^{(k)} = \text{BiGRU}(x^{(k)}). \tag{1}$$

Subsequently, unlike Castro Ferreira et al. (2018a), we encode $h^{(k)}$ into the context representation $c^{(k)}$ using self-attention (Yang et al., 2016). Specifically, given the total N steps in $h^{(k)}$, we first calculate the attention weight $\alpha_j^{(k)}$ at each step j by:

$$e_j^{(k)} = v_a^{(k)T} \tanh(W_a^{(k)} h_j^{(k)}), \tag{2}$$

$$\alpha_j^{(k)} = \frac{\exp(e_j^{(k)})}{\sum_{n=1}^{N} \exp(e_n^{(k)})}, \tag{3}$$

where v_a is the attention vector and W_a is the weight in the attention layer. The context representation of $x^{(k)}$ is then the weighted sum of $h^{(k)}$:

$$c^{(k)} = \sum_{j=1}^{N} \alpha_j^{(k)} h^{(k)}. \tag{4}$$

After obtaining $c^{(pre)}$ and $c^{(pos)}$, we concatenate them with the target entity embedding $x^{(r)}$, and pass it through a feed-forward network to obtain the final representation:

$$R = \text{ReLU}(W_f[c^{(pre)}, x^{(r)}, c^{(pos)}]), \tag{5}$$

where W_f is the weights in the feedforward layer. R is also used as the input of the probing classifiers (Section 6). R is then fed for making the final prediction:

$$P(f|x^{(pre)}, x^{(r)}, x^{(pos)}) = \text{Softmax}(W_c R), \tag{6}$$

where W_c is the weight in the output layer.

C-RNN-EL. In addition to ConATT-EL, we also try a simpler yet effective structure, which uses only a single BiGRU. We name the framework it follows as the centred recurrent neural networks (henceforth c-RNN). Specifically, instead of using two separate BiGRUs to encode pre- and pos-contexts, we first concatenate $x^{(pre)}$, $x^{(r)}$, and $x^{(pos)}$, and then encode them together:

$$h = \text{BiGRU}([x^{(pre)}, x^{(r)}, x^{(pos)}]). \tag{7}$$

Assuming the target entity is at position i of the concatenated sequence, we extract the ith representation from h_i to obtain $R = \text{ReLU}(W_f h_i)$. After obtaining R, the rest of the procedure is the same as ConATT-EL.

Pre-training. One of our aims (see Section 3.1) is to find out whether RFS can benefit from pre-trained word embeddings and language models, whose effectiveness for REG has not yet been investigated. For both c-RNN-EL and ConATT-EL, we try the GloVe embeddings[7] (Pennington et al., 2014) for English and SGNS embeddings for Chinese[8] (Li et al., 2018) to see how pre-trained word embeddings contribute to the choice of RF.

For c-RNN-EL, we also try to stake it on the BERT (Devlin et al., 2019) model. To make BERT better encode the delexicalised entity labels for English, we first re-train BERT as a masked language model on the training data of WEBNLG. We then freeze the parameters of BERT and use the model to encode the input, which is then fed into c-RNN-EL.[9] We name this re-trained BERT as BERT-RT. For Chinese, however, all Chinese BERT models are character-based, which means that each entity label becomes a sequence of characters. Therefore, this method is not applicable to c-RNN-EL, since this model assumes that each entity label is a single token. Therefore, we do not use BERT on Chinese c-RNN-EL.

5.2. RFS-PN models

In contrast to RFS-EL models, RFS-PN models take a different form of input. The target referent $x^{(r)}$ is a sequence of words instead of a single label, and therefore, these models need different mechanisms for encoding $x^{(r)}$.

ConATT-PN. After obtaining $c^{(pre)}$ and $c^{(pos)}$ from $x^{(pre)}$ and $x^{(pos)}$ using two self-attentions (Eqs. (1)–(4)), ConATT-PN uses another self-attention to encode $x^{(r)}$ and obtain $c^{(r)}$. Subsequently, we compute the final representation by adapting Eq. (5) to

$$R = \text{ReLU}(W_f[c^{(pre)}, c^{(r)}, c^{(pos)}]). \tag{8}$$

C-RNN-PN. Similar to c-RNN-EL, c-RNN-PN first concatenates $x^{(pre)}$, $x^{(r)}$, and $x^{(pos)}$, and encodes them together to obtain h using Eq. (7). Since the target referent has multiple words, we use the summation of the hidden representations at the beginning and end of the target referent (i.e., i and j) for calculating the final representation:

$$R = ReLU(W_f[h_i + h_j]). \tag{9}$$

Pre-training. For both ConATT-PN and c-RNN-PN, we try GloVe for English and SGNS for Chinese. For c-RNN-PN, we try BERT for both English and Chinese.

Table 6
Features used in the XGBoost models.

Feature	Definition	WEBNLG	ONTO-EN	ONTO-ZH
Syn	See Section 6.2.	✓	✓	✓
Entity	Person, Organisation, Location, Other	✓	✓	–
Gender	Values: male/female/other	✓	–	–
DisStat	See Section 6.2.	✓	✓	✓
SenStat	See Section 6.2.	✓	✓	✓
DistAnt	See Section 6.2.	✓	✓	✓
IntRef	See Section 6.2.	✓	✓	✓
DistAnt_W	Distance in number of words (5 quantiles)	✓	–	–
Sent_1	Does RE appear in the first sentence?	✓	–	–
MetaPro	Description see Section 6.2.	✓	–	–
GloPro	Description see Section 6.2.	✓	✓	✓

Table 7
Evaluation results of our RFS systems on WEBNLG. Best results are **boldfaced**, whereas the second best results are underlined.

	4-way			3-way			2-way		
Model	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
XGBoost	53.77	51.98	51.55	71.27	69.24	68.34	86.64	82.76	84.57
c-RNN-EL	68.79	62.95	64.96	84.49	82.52	**83.63**	90.31	88.01	89.09
+GloVe	**69.10**	**63.90**	**65.40**	84.29	**82.55**	83.30	89.33	88.02	88.63
+BERT-RT	62.63	61.80	62.15	83.02	81.44	82.15	**90.98**	88.00	**89.42**
ConATT-EL	67.42	62.39	64.07	**85.04**	82.21	83.53	89.30	**89.19**	89.23
+GloVe	65.98	62.49	63.67	83.62	81.41	82.45	89.60	88.06	88.80
c-RNN-PN									
+BERT	67.19	58.81	61.08	81.47	80.16	80.54	86.83	88.41	87.60

5.3. Baseline

We use a Machine Learning (ML) model as our baseline for both RFS-EL and RFS-PN.[10] We used XGBoost (Chen and Guestrin, 2016) from the family of Gradient Boosting Decision Trees to train RFS classifiers. A 5-fold-cross-validation was used to train the models. The models are mainly trained on the features defined for the probing tasks. Additionally, some of the models include further features, such as entity type and gender. Table 6 shows the full list of the features used.

5.4. Implementation details and evaluation protocols

We tuned the hyper-parameters of each of our models on the development set of each dataset and chose the setting with the best macro F1 score. For the BERT model, we used the BERT-BASE-CASED[11] for English and the BERT-BASE-CHINESE[12] for Chinese. For BERT-RT, we re-trained BERT on WEBNLG. We set the masking probability to 0.15 and trained it for 25 epochs. For the XGBoost models, we set the learning rate to 0.05, the minimum split loss to 0.01, the maximum depth of a tree to 5, and the sub-sample ratio of the training instances to 0.5.

The Chinese BERT is character-based (i.e., the inputs are broken down into characters), while all English models are word-based. When we test RFS-PN on ONTONOTES-ZH$_{\leq 512}$, we report the performance of both character-based and word-based models to show that our conclusions are not affected by whether the inputs are characters or words. We run each model 5 times and report the macro-averaged precision, recall, and F1 on the test set.

5.5. Results on WEBNLG

Table 7 shows the results of the different classification tasks on WEBNLG in English. All neural variants outperform the machine learning baseline. The difference in performance is small for binary classification, but much larger for 3-way and 4-way classification. As the 2-way classification task (i.e., pronominalisation) is simpler than the other two classifications, the feature set used by the baseline produces almost similar results as the neural models.

[7] https://nlp.stanford.edu/projects/glove/

[8] https://github.com/Embedding/Chinese-Word-Vectors

[9] We also explored other ways of using BERT, such as using only BERT plus a feed-forward layer to obtain h, or not freezing the parameters of BERT during training. The resulting models had low performance in all cases.

[10] This model is feature-based and the features used are the same in both RFS-EL and RFS-PN.

[11] huggingface.co/bert-base-cased

[12] huggingface.co/bert-base-cased

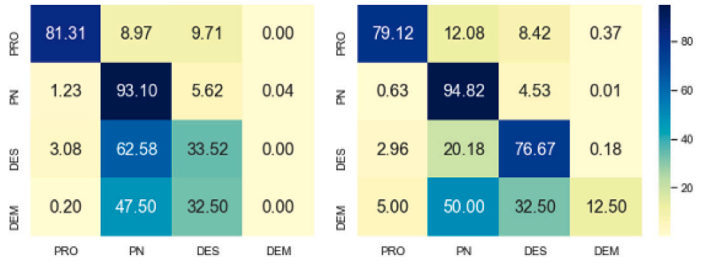

Fig. 1. Confusion Matrices for 4-way classification results of `XGBoost` (left) and `c-RNN-EL+GloVe` (right), where PRO, PN, DES, and DEM are pronoun, proper name, description and demonstrative, respectively. The vertical axis indicates the reference labels while the horizontal axis indicates the predicted labels.

Comparing the neural variants, the results show that the simpler `c-RNN-EL` wins over `ConATT-EL` in 4-way classification and performs on-par with `ConATT-EL` in 3- and 2-way classification. This is against $\mathcal{E}_1$, which expects that `ConATT-EL` outperforms `c-RNN-EL`. One possible explanation is that `ConATT-EL` first breaks down the input into three pieces (i.e., the target entity and the pre- and pos-contexts), encodes them separately, and then merges the encoded representations back together before sending them to make predictions. This "divide and merge" procedure might hinder the model from learning some useful information.

Regarding the effectiveness of incorporating pre-trained models, `GloVe` embeddings have a positive impact on `c-RNN` only for 4-way predictions and do not contribute to 2- and 3-way classifications. Moreover, it has a negative impact on `ConATT`: performance decreases when `GloVe` is used.

Concerning `BERT`, a surprising observation is that in the case of `c-RNN-EL`, `BERT-RT` has a negative effect on the 4-way and 3-way predictions (F1 score decreases from 64.86 and 83.63 to 62.15 and 82.15, respectively). In the pronominalisation task, `BERT` slightly increases performance (from 89.09 to 89.42), but this increase is not as much as `BERT`'s boosting effect on other NLP tasks. This is probably because although `BERT` has been re-trained on WEBNLG delexicalised sentences, the entity labels still act as noise for `BERT`.

To rule out the above speculation, we also tried out the models without using entity labels. We tested `c-RNN-PN` with `BERT` on WEBNLG, but `BERT` still had a negative effect. This result is in contrast to $\mathcal{E}_2$, which assumes that pre-trained models would contribute positively to RFS. Expectations $\mathcal{E}_1$ and $\mathcal{E}_2$ on WEBNLG could be rejected in part because the dataset itself is overly formal and therefore uses REs in a simplified way. As mentioned earlier, 85% of the REs in WEBNLG are first mentions and 71% of the REs are proper names. Therefore, models with complex architecture or pre-trained models may not be able to show their true strengths.

To gain insight into the behaviour of the deep learning and classic ML-based models for RFS, we plot in Fig. 1 the confusion matrices of `XGBoost` and the best performing neural model `c-RNN-EL+GloVe` for 4-way classification. The confusion matrices show that both models perform well in predicting pronouns and proper names (hence the difference in performance is small for 2-way classification) and both perform poorly in predicting demonstratives (probably due to the fact that demonstratives are extremely infrequent in WEBNLG).

The main difference between the two models lies in distinguishing proper names from descriptions. The `XGBoost` model incorrectly predicted descriptions as proper names 62.58% of the time, while the neural model `c-RNN-EL+GloVe` made this incorrect prediction 20.18% of the time. This difference in the performance of the two models could be due to the fact that the neural models have learned some useful features from the discourse that are not captured in our feature engineering procedure.

In addition, when we examined the WEBNLG dataset, we found that several RE cases are incorrectly annotated. For example, WEBNLG annotates "*United States*" as a proper name and "*the United States*" as a description. The incorrect annotations could add to the confusion between the choice of description and proper name in both `XGBoost` and `c-RNN-EL+GloVe`.

5.6. *Results on* ONTONOTES

Table 8 and 9 show the performance of our RFS models on English and Chinese ONTONOTES, respectively. In the case of Chinese, we tested our models on both ONTONOTES and ONTONOTES-ZH$_{\leq 512}$, using both character-based and word-based input formats. We provide the results of our word-based RFS-PN models on ONTONOTES-ZH$_{\leq 512}$ in Appendix A to show that the input format (comparing it to the character-based RFS-PN models on ONTONOTES-ZH$_{\leq 512}$) and sub-sampling (comparing it to the word-based RFS-PN models on ONTONOTES-ZH) do not influence the results as much.

Results in English. For the English portion, we find that, in line with our expectation $\mathcal{E}_3$, the performance of the RFS-EL models is lower than their performance on WEBNLG. Looking at the results on ONTONOTES-EN, it is surprising that for 2-way classification, the baseline (i.e., the data-driven, feature-based model) defeats almost all neural models except `c-RNN-PN+BERT`. For 4-way and 3-way classification, it can still beat the RFS-EL models but performs worse than the RFS-PN models.

Consistent with the experiments on WEBNLG, the simpler `c-RNN` models (either `EL` or `PN`) outperform or are at least on par with the `ConATT` models on ONTONOTES-EN. Furthermore, pre-trained word embeddings (i.e., `GloVe`) make a positive contribution to both RFS-EL and RFS-PN, and `BERT` significantly improves performance on all classification tasks. For example, if we compare `c-RNN-PN+BERT` with `c-RNN-PN` for the full RFS-PN task (i.e., 4-way classification), `c-RNN-PN+BERT` improves performance (F1 score) from 62.38 to 74.59.

Table 8
Evaluation results of our RFS-EL and RFS-PN systems on ONTONOTES-EN. Each percentage below the F-score of c-RNN-PN+BERT indicates how much c-RNN-PN gains from using BERT compared to not using BERT.

Model	4-way			3-way			2-way		
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
XGBoost	48.96	49.69	49.12	67.78	65.78	66.44	79.11	78.01	78.42
c-RNN-EL	50.77	45.89	46.38	60.83	59.56	59.94	73.33	72.58	72.84
+Glove	53.47	49.49	50.44	61.72	60.66	60.98	75.06	73.96	74.32
ConATT-EL	52.32	45.88	46.89	59.66	58.71	59.08	71.86	71.38	71.56
+Glove	54.55	47.56	48.14	59.75	60.05	59.85	73.84	72.32	72.66
c-RNN-PN	65.45	60.59	62.38	68.19	69.19	68.55	76.66	75.23	75.70
+Glove	66.06	63.39	64.56	69.94	70.14	70.01	77.61	76.31	76.67
+BERT	**73.57**	**75.94**	**74.59** (+19.57%)	**80.53**	**81.81**	**81.03** (+18.21%)	**87.21**	**86.97**	**87.08** (+15.03%)
ConATT-PN	61.29	62.21	61.58	66.34	65.87	66.01	73.19	73.21	73.19
+Glove	63.71	61.70	62.51	67.18	66.88	67.00	75.17	74.48	74.75

Table 9
Evaluation results of our word-based RFS systems on the ONTONOTES-ZH dataset and character-based RFS systems on the ONTONOTES-ZH$_{\leq 512}$ dataset. The subscript "w" indicates that the model takes words as inputs and "c" indicates that the model takes characters as inputs.

	5-way			4-way			3-way			2-way		
Model	P	R	F	P	R	F	P	R	F	P	R	F
ONTONOTES-ZH												
XGBoost	38.17	40.06	34.59	46.16	44.12	41.29	56.19	54.64	51.98	64.5	79.56	63.67
c-RNN$_w$-EL	45.34	43.27	43.62	53.55	51.51	52.18	56.11	53.33	54.42	64.91	63.64	64.22
+SGNS	51.13	48.13	48.05	59.14	57.65	57.63	59.15	55.46	56.78	66.76	68.57	67.58
ConATT$_w$-EL	46.01	43.79	44.28	53.69	52.93	53.05	55.25	54.04	54.55	64.60	65.85	65.01
+SGNS	50.78	47.77	47.84	57.75	55.98	56.42	59.34	55.40	56.87	67.04	68.30	67.59
c-RNN$_w$-PN	52.36	47.91	48.97	54.14	52.40	53.06	55.30	52.99	53.86	64.88	62.81	63.68
+SGNS	56.67	53.82	54.30	59.38	57.40	58.23	59.58	56.66	57.78	67.75	66.28	66.91
ConATT$_w$-PN	50.41	45.45	46.86	51.27	49.80	50.35	59.06	54.43	56.11	63.71	63.75	63.73
+SGNS	52.33	48.60	49.37	53.48	51.64	52.38	60.53	56.18	57.69	67.86	64.97	65.95
ONTONOTES-ZH$_{\leq 512}$												
c-RNN$_c$-PN	52.42	48.49	49.62	54.60	54.65	54.19	56.78	53.50	54.68	67.66	62.89	64.59
+SGNS	54.54	51.27	51.56	57.78	56.75	57.16	59.57	56.19	57.46	67.74	65.33	66.37
+BERT	**64.99**	**63.60**	**63.85** (+28.68%)	**68.22**	**69.48**	**68.17** (+25.80%)	**70.36**	**68.60**	**69.13** (+26.43%)	**78.35**	**73.51**	**75.59** (+17.03%)
ConATT$_c$-PN	51.78	48.28	49.25	54.27	53.08	52.98	53.67	49.47	50.79	63.25	56.92	58.28
+SGNS	55.44	52.13	53.09	55.88	54.94	54.18	55.01	53.06	53.87	64.98	61.38	62.69

Results in Chinese. As for the results on the Chinese portion, unlike the English results, the baseline defeats very few neural RFS models in the 2-way classification (i.e., choosing between ZP and overt RE) and only ConATT$_c$-PN in the 3-way classification. For the 5-way and 4-way classification, all neural models significantly beat the baseline. For example, the best model c-RNN$_c$-PN+BERT achieves an F1 score of 63.85 in the 5-way classification, which is far higher than the 34.59 of the baseline. This is probably because the Chinese RFS task has a higher complexity than the English RFS, which we will discuss later in this section.

With the exception of the 5-way classification of ONTONOTES-ZH$_{\leq 512}$, c-RNN models generally perform better or at least similar to ConATTs. The pre-trained word embeddings again help with all kinds of RFS tasks. Similar to English, Chinese RFS benefits enormously from the use of BERT. For example, in 5-way classification, it improves performance by about 20% compared to the second best model (i.e., ConATT$_c$-PN+SGNS).

Analysis of expectations. Overall, our first expectation $\mathcal{E}_1$, (i.e., ConATT works better than c-RNN) is incorrect with respect to both languages. The second expectation $\mathcal{E}_2$ (i.e., models using pre-trained word embeddings perform better than models without pre-training, and worse than models with pre-trained contextual language models) is partially rejected in English, as BERT sometimes has negative effects, but it is confirmed in Chinese.

The fourth expectation $\mathcal{E}_4$ is confirmed, as models with proper names as referent representations always perform better than their counterparts with entity labels. More specifically, on ONTONOTES-EN, the RFS-PN models perform significantly better than the RFS-EL models. For example, c-RNN-PN+GloVe has an F1 score of 64.56, while c-RNN-EL+GloVe's F1 score is only 50.44. On ONTONOTES-ZH, this difference is smaller, partly due to the fact that Chinese proper names more often consist of only one word; therefore, Chinese entity labels are more likely to match the corresponding proper names.

In line with our expectation $\mathcal{E}_5$, models that work well in English also work well in modelling ZP in Chinese, but deciding whether to use a ZP or an overt RE is more difficult than pronominalisation. For example, c-RNN-PN achieved an F-score of 75.7 for the

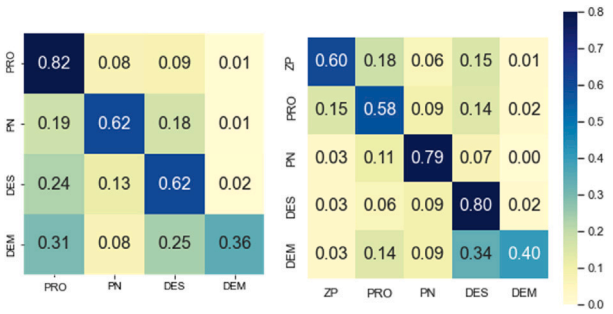

Fig. 2. Confusion Matrix for English 4-way `c-RNN-PN+BERT` (left) and Confusion Matrix for Chinese 5-way `c-RNN`$_c$`-PN+BERT` (right) on ONTONOTES.

English 2-way task, while the performance of `c-RNN`$_c$`-PN` was only 64.59 for Chinese. The confusion matrices (Fig. 2) suggest that both English and Chinese RFS models confuse demonstratives with descriptions.

The results of both Chinese and English RFS tasks improve dramatically when the contextual language model `BERT` is used. To test our last hypothesis $\mathcal{H}_6$, we compute how much `c-RNN-PN` gains from using `BERT` compared to not using `BERT` and report the numbers in Tables 8 and 9. On average, `c-RNN-PN` gains 17.60% from using `BERT` in English and 24.48% in Chinese. The results suggest that Chinese RFS benefits more from `BERT` than English RFS. Nevertheless, we cannot make conclusive statements about $\mathcal{H}_6$. Strictly speaking, these percentages are not directly comparable and the comparison cannot be fully controlled because, for example: (1) the data is not fully parallel, and (2) the RFS tasks defined for the two languages differ from each other. For instance, Chinese RFS, unlike English RFS, considers an additional category, namely ZP.

5.7. Summary

We introduced several RFS models in this section and evaluated them on WEBNLG and ONTONOTES. In short, we found that (1) the models that worked well on WEBNLG had lower performance on ONTONOTES, probably because the ONTONOTES data is multi-genre and more complex. (2) The models which worked well in English also worked well in Chinese, where ZPs are frequently used. Chinese RFS models benefit more from contextual pre-trained language models than English RFS models, but such a statement is not conclusive. (3) Representing entities using their proper names instead of their entity labels helped the models to better deal with unseen entities.

6. Probing RFS models

Having compared the performance of the RFS models on both corpora, it is now time to investigate what each model has learned.

6.1. Probing classifiers

We use a logistic regression classifier as our probing classifier. Concretely, for each input, we first use a model discussed in Section 5 to obtain its representation R. As mentioned in Section 5, we ran each model five times and reported their averaged scores. For the probing tasks, we use the representations of the models with the best RFS performance on the development set.

6.2. Probing tasks

Following our discussion in Section 2.2, we formulate the following probing tasks.

Referential status. Both linguistic (Chafe, 1976; Gundel et al., 1993) and computational studies (Castro Ferreira et al., 2016) have examined the role of referential status as one of the factors influencing the choice of RF. We define referential status at two levels: discourse-level and sentence-level. The former (**DisStat**) has two possible values: (a) discourse-old (i.e., the entity has appeared in the previous discourse) and (b) discourse-new (the entity is new in the discourse). The sentence-level referential status (**SenStat**) also consists of two values: (a) sentence-new (the RE is the first mention of the entity in the sentence), and (b) sentence-old (the RE is not the first mention of the entity in the sentence).

Syntactic position. Entities in the subject position are more likely to be pronominalised than those in the object position (Brennan, 1995; Arnold, 2010). Therefore, in the syntax probing task (**Syn**), we do a binary classification: subject or object.

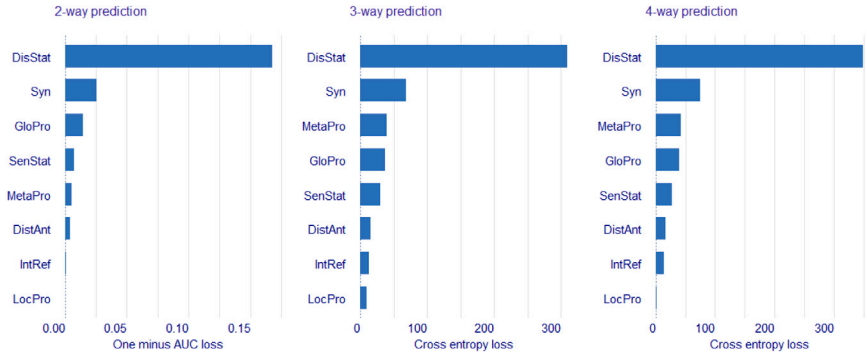

Fig. 3. Feature importance of the XGBoost classifiers for predictions on WEBNLG. Higher loss shows greater importance of a feature.

Recency. Recency is a key feature in many of the previous REG and RFS models (Greenbacker and McCoy, 2009; Kibrik et al., 2016). It measures the distance between the target entity and its closest coreferring antecedent. There are various ways of estimating the recency of a target entity given its context. We use two measures here. The first one (**DistAnt**) measures the number of sentences between the target entity and its antecedent, and has four possible values: the entity and its antecedent are (a) in the same sentence, (b) one sentence away, (c) more than one sentence away, and (d) the entity is mentioned for the first time in the discourse (to distinguish first mentions from subsequent mentions). The second measure (**IntRef**) asks whether there is an intervening referent between the target and its nearest antecedent. In other words, it checks whether the target and the preceding RE are coreferential. This feature has three possible values: (a) the target entity is a first mention, (b) the preceding RE refers to the same entity, and (c) the preceding RE refers to a different entity. Note that the presence of intervening markables might signal the existence of competition (i.e., the intervening referent has the same animacy and gender values as the target RE).

Discourse structure prominence. As mentioned in Section 2, the "organizational" properties of discourse may influence the prominence status of the entities. We introduce three probing tasks capturing different properties of the discourse. (1) *Local prominence* (**LocPro**): The idea of local prominence comes from Centering Theory (Grosz et al., 1995) and it is a hybrid feature of DisStat and Syn. Concretely, we use the implementation of Henschel et al. (2000): an entity is *locally prominent* if it is "discourse-old" and "realised as subject". It is a binary feature with two possible values: (a) locally prominent, and (b) not locally prominent. (2) *Global prominence* (**GloPro**): This feature is based on the notion of global salience in Siddharthan et al. (2011), asking whether the entity is a minor or major referent in the text. According to them, "the frequency features are likely to give a good indication of the global salience of a referent in the document" (Siddharthan et al., 2011, p. 820). We define a binary feature in which the most frequent entity in a text is marked as globally prominent. (3) *Meta-prominence* (**MetaPro**): In line with global prominence, we also want to explore to what extent prominence beyond a single text (e.g., on a text collection level) may impact the way people refer. In the context of the current circumstances, the sentence "I received *my vaccine* today" is unambiguous, and the RE *my vaccine* needs no extra modification (e.g. my COVID-19 vaccine); however, a couple of years from now, a richer RE may be needed to refer to the vaccine. The idea behind this exploratory feature is that people might use less semantic content to refer to the referents which are well known outside of the text. Based on the number of mentions of a target entity in the whole corpus, four possible values, each of which representing an interval, are assigned to each RE: (a) $[0, 50)$, (b) $[50, 150)$, (c) $[150, 290)$, and (d) $[290, \infty)$. For example, the category $[0, 50)$ contains those entities that occur fewer than 50 times in the corpus. Note that since, different from WEBNLG, ONTONOTES was collected from a wide range of resources, there are very few referents appearing more than 50 times in the corpus. We, therefore, consider probing MetaPro merely on WEBNLG.

6.3. Importance analysis

We conducted a feature importance analysis to determine which of the features used in the probing tasks contributed most to the feature-based ML models. This analysis serves as a sanity check to find out whether the representations have learned the features that contribute most to the RFS task.

To assess the importance of the features used in the probing tasks, we train XGBoost models using only features from Section 6.2, and calculate the model-agnostic permutation-based variable importance of each model (Biecek and Burzykowski, 2021). Specifically, we measure the extent to which performance changes when we remove one of the features. Fig. 3 shows the change in performance for each feature on WEBNLG.

According to the figure, DisStat and Syn contribute the most, while LocPro is the least important feature, being a hybrid combination of DisStat and Syn. This means that if we remove this feature while keeping DisStat and Syn, the performance of the model will not be significantly affected. Considering that DisStat and Syn are both very important features, LocPro is much more important than the experiment suggests. In addition to the DisStat and Syn probing tasks, we, therefore, expect high performance for LocPro. The results of the importance analysis on ONTONOTES can be found in Appendix B.

Table 10
Macro-averaged F1 scores of each probing task on WEBNLG, where "m_avg" means average value per model per task and "t_avg" means average value per task.

Model	Type	DisStat	SenStat	Syn	DistAnt	IntRef	LocPro	GloPro	MetaPro
Random	–	41.83	22.87	48.99	14.90	22.92	49.84	48.02	25.20
majority	–	46.50	31.00	37.99	23.25	31.00	36.01	40.65	10.97
c-RNN	4-way	84.06	73.72	85.34	53.84	55.43	82.92	56.00	42.32
	3-way	83.72	72.60	83.60	54.78	53.21	81.67	56.70	41.79
	2-way	88.04	73.84	84.00	54.93	52.31	85.69	59.98	41.65
	m_avg	85.27	73.39	84.31	54.52	53.65	83.43	57.56	41.92
c-RNN +GloVe	4-way	84.85	74.59	87.04	55.67	55.93	83.20	53.53	41.71
	3-way	83.89	67.24	82.48	50.94	51.17	81.44	52.49	42.34
	2-way	88.02	71.25	82.67	53.67	51.43	85.22	63.17	41.03
	m_avg	85.59	71.03	84.06	53.43	52.84	83.29	56.40	41.69
c-RNN +BERT	4-way	90.64	78.04	82.71	56.91	54.30	81.67	54.24	43.07
	3-way	84.80	72.29	84.08	54.21	53.25	82.53	57.31	42.80
	2-way	87.28	69.69	84.74	54.19	54.88	82.77	63.07	40.75
	m_avg	87.57	73.34	83.84	55.10	54.14	82.32	58.21	42.21
ConATT	4-way	87.81	77.11	88.00	57.09	55.88	86.34	60.15	46.14
	3-way	84.39	74.19	86.66	55.26	54.09	84.56	60.61	47.47
	2-way	84.20	73.18	88.44	53.98	53.64	86.75	56.39	41.81
	m_avg	85.47	74.83	87.7	55.44	54.54	85.88	59.05	45.14
ConATT +GloVe	4-way	87.82	77.70	87.24	57.52	55.22	85.69	58.54	49.94
	3-way	84.35	72.83	88.91	54.23	51.96	86.80	59.05	46.36
	2-way	84.38	73.21	86.96	56.14	53.33	85.27	62.46	39.63
	m_avg	85.52	74.58	87.70	55.96	53.50	85.92	60.02	45.31
–	t_avg	85.88	73.43	85.52	54.89	53.74	84.17	58.25	43.25

6.4. Baselines and evaluation protocols

We evaluate probing tasks using the macro-averaged F1 scores. We also report the accuracy of each probing task in C. We train each probing classifier 5 times and report the average value. To better describe the results, we also report the average performance per model per task and the average performance per task.

We use 2 baselines: (1) `random`: it randomly assigns a label to each input; and (2) `majority`: it assigns the most frequent label in the given probing task to the inputs. In this section, we will not probe all the models from Section 5. Specifically, we probe only the RFS-PN models for ONTONOTES.

6.5. Probing experiments

As mentioned earlier, we perform probing tasks to find out whether the latent representations of the RFS models encode the features mentioned in Section 6.2. High performance on the probing tasks would indicate that the features are encoded in the latent representations of the models.

Results on WEBNLG. Table 10 shows the results of probing experiments on WEBNLG. Compared to the `random` baseline, all neural models have achieved higher performance on all tasks. Concretely, we made the following observations:

1. Referential status and syntactic position: all models exhibit consistently high performance on DisStat, SenStat, and Syn. This shows that all neural models can learn information about referential status and syntactic position;
2. Recency (i.e., DistAnt and IntRef): all models perform worse than the referential status and syntax probes. Their F1 scores are lower than those of DisStat, SenStat, and Syn, and are closer to the baselines. This finding is consistent with the importance analysis in Section 6.3, where DistAnt and IntRef were found to be less important than DisStat and Syn. This may be due to the fact that 67% of the documents in the WEBNLG corpus only contain one sentence, making recency-related features less relevant. It is also possible that the models have greater difficulty capturing long-distance properties, in line with previous probing works on co-reference and bridging anaphora (Sorodoc et al., 2020; Pandit and Hou, 2021);
3. Discourse structure prominence: since LocPro is a hybrid of DisStat and Syn, all models were able to handle it quite well. Meanwhile, the models seem to handle GloPro and MetaPro worse than the other features since the performance of the corresponding probing tasks is closer to the baselines.[13] These results are in contrast to the results of the importance analysis, which suggests that both GloPro and MetaPro are important features (ranked 3 and 4 in Fig. 3). Learning GloPro and MetaPro requires a model to have an overall understanding of the entire input document or corpus, which the neural models might not be able to acquire.

[13] Note that, for MetaPro, the `Majority` has a low F1 score, as the distribution of the values of MatePro is balanced.

Table 11
Results of our baselines and RFS-PN models on each probing task on the ONTONOTES-EN dataset.

Model	Type	DisStat	SenStat	Syn	DistAnt	IntRef	LocPro	GloPro
Random	–	49.77	32.27	50.10	23.75	32.40	48.21	49.53
majority	–	35.88	20.39	33.39	15.29	20.39	40.50	38.68
c-RNN-PN	4-way	63.39	50.76	74.67	36.31	44.81	67.86	50.32
	3-way	63.30	50.45	75.55	36.78	42.83	68.26	49.71
	2-way	63.31	49.72	73.82	35.75	43.03	65.72	45.76
	m_avg	63.33	50.31	74.68	36.28	43.56	67.28	48.60
c-RNN-PN +GloVe	4-way	64.24	51.39	76.75	37.09	44.94	67.26	51.44
	3-way	64.44	52.69	78.06	37.55	45.89	70.66	53.28
	2-way	64.26	49.34	75.06	35.87	45.04	67.22	47.49
	m_avg	64.31	51.14	76.62	36.84	45.29	68.38	50.74
c-RNN-PN +BERT	4-way	85.67	69.46	79.73	50.36	65.99	80.08	60.06
	3-way	83.42	68.90	81.15	49.10	63.62	82.38	61.93
	2-way	81.12	67.07	77.89	47.97	62.06	77.45	53.37
	m_avg	83.40	68.48	79.59	49.14	63.89	79.97	58.45
ConATT-PN	4-way	62.95	46.63	73.34	33.33	43.52	66.30	48.89
	3-way	61.87	45.92	74.76	31.91	41.64	67.51	48.61
	2-way	59.46	41.73	63.72	30.18	40.85	59.96	47.51
	m_avg	61.43	44.76	70.61	31.81	42.00	64.59	48.34
ConATT-PN +GloVe	4-way	63.41	50.49	79.95	36.03	43.17	70.27	49.86
	3-way	61.79	45.39	79.00	33.03	41.53	68.46	48.97
	2-way	61.56	44.35	73.97	31.53	42.81	63.31	48.39
	m_avg	62.25	46.74	77.64	33.53	42.50	67.35	49.07
–	t_avg	64.67	48.94	72.81	35.53	45.04	67.15	50.05

Table 12
Results of RFS-PN models on each probing task on the ONTONOTES-ZH$_{\leq 512}$ dataset.

Model	Type	DisStat	SenStat	Syn	DistAnt	IntRef	LocPro	GloPro
Random	–	49.93	32.70	49.79	23.81	33.01	46.44	44.27
majority	–	36.43	19.95	36.62	14.96	19.95	43.27	45.09
c-RNN$_c$-PN	5-way	62.80	45.89	75.94	28.49	45.78	65.54	52.03
	4-way	61.80	43.39	74.74	27.73	44.65	63.44	46.64
	3-way	61.19	41.52	76.11	26.43	41.13	61.70	45.76
	2-way	58.06	36.30	76.96	24.11	36.49	58.82	45.54
	m_avg	60.96	41.78	75.94	26.69	42.01	62.38	47.49
c-RNN$_c$-PN +SGNS	5-way	63.52	47.24	77.28	30.71	46.13	66.11	50.37
	4-way	62.90	46.96	77.18	30.78	47.81	66.16	48.20
	3-way	62.87	42.54	77.81	27.51	43.59	64.17	46.11
	2-way	60.45	38.24	77.12	24.27	37.61	64.09	46.12
	m_avg	62.44	43.75	77.35	28.32	43.79	65.13	47.70
c-RNN$_c$-PN +BERT	5-way	75.20	57.07	78.68	39.54	57.69	70.93	55.17
	4-way	73.96	57.66	78.15	37.12	56.90	69.68	46.60
	3-way	73.77	56.29	79.67	35.77	55.96	73.24	45.59
	2-way	68.10	52.08	79.84	29.71	52.36	71.30	45.07
	m_avg	72.76	55.78	79.09	35.54	55.73	71.29	48.11
ConATT$_c$-PN	5-way	62.33	43.17	73.94	28.92	45.05	63.99	47.16
	4-way	61.91	43.15	67.48	26.41	44.15	57.31	47.27
	3-way	59.54	39.55	68.78	24.47	39.13	55.27	45.73
	2-way	52.10	32.85	65.67	21.78	32.66	49.38	45.35
	m_avg	58.97	39.68	68.97	25.40	40.25	56.49	46.38
ConATT$_c$-PN +SGNS	5-way	63.48	45.60	77.27	29.77	46.84	64.16	50.72
	4-way	61.97	44.63	74.65	28.19	46.61	64.49	47.27
	3-way	58.79	38.78	74.09	24.66	38.51	60.19	45.73
	2-way	60.09	39.53	72.90	22.13	34.88	61.43	45.35
	m_avg	61.08	42.14	74.73	26.19	41.71	62.57	47.27
–	t_avg	63.24	44.62	75.21	28.43	44.70	63.57	47.39

Results on ONTONOTES. Table 11 and 12 report the results on ONTONOTES-EN and ONTONOTES-ZH$_{\leq 512}$, respectively. The general observations on these two subcorpora are as follows:

1. Compared to the WEBNLG probing results, the scores for ONTONOTES-EN are generally lower. This is consistent with the RFS results (i.e., RFS models also have lower performance on ONTONOTES-EN than on WEBNLG) and follows from the fact that the data in ONTONOTES is more complex than the data in WEBNLG;
2. Focusing on ONTONOTES, all models for both languages can learn a certain amount of information about all features except GloPro;
3. Except BERT-based models, all models perform similarly on the GloPro task. Using BERT can help the model learn slightly more GloPro information, but these improvements are much smaller than those obtained in the other probing tasks. Two possible explanations could be given for this. One reason is that, as mentioned earlier, neural models are not good at counting how many times a referent occurs in the discourse and are therefore unable to pick out the dominant referents. The other reason is that we created each input in ONTONOTES using only 3 sentences before the target referent and 3 sentences after the target referent. This sometimes leads to the dominant referent occurring only once in a given discourse. Thus, for each input in the current format, there is little or no difference between the frequency of the prominent referent and that of other referents, which hinders the classifiers from distinguishing them;
4. Similar to WEBNLG, all models perform remarkably well on the tasks DisStat, Syn, and LocPro. This suggests that all models have learned information about the referential status and grammatical role of the target referents. Since LocPro is a hybrid of DisStat and Syn, it is no surprise that our models can handle it well;
5. The performance of SenStat, DistAnt, and IntRef is lower than that of the three tasks above. This drop in performance is understandable because learning these features requires a model to not only check whether the target referent occurs in the pre-context, but also to roughly locate it in that context. These tasks are clearly more demanding than DisStat and Syn. All models have the lowest performance on the DistAnt task. This is partly because, compared to SenStat and IntRef, this task asks each model to locate the previous mention of the target referent in a more fine-grained way: It checks whether the previous mention occurs in the current sentence, in the previous sentence or further away.

Comparing c-RNN and ConATT. We concluded in Section 5 that compared to the ConATT models, the c-RNN models can sometimes perform better or at least equally well on both datasets (WEBNLG and ONTONOTES) and languages (English and Chinese). However, the probing results are different for the two datasets.

Unlike the RF classification results, for WEBNLG, we find that ConATT performs better than c-RNN on many tasks, including DisStat (They receive similar F1 scores, but ConATT achieves much better accuracy. See Table C.14 for more details.), LocPro, GloPro, and MetaPro. In contrast, for ONTONOTES, c-RNN learns significantly more information about syntactic position and slightly more information about referential status (i.e., SenStat) and recency (i.e., IntRef) than ConATT, which is consistent with c-RNN winning at RFS classification. In summary, our expectation $\mathcal{E}_7$, that the better performing models would learn more relevant linguistic information, is rejected for the WEBNLG corpus but confirmed for the ONTONOTES corpus. This is probably due to the fact that the REs in WEBNLG are not representative of the realistic use of REs (see Section 4 for further discussion).

The effect of pre-training. Recall that for the RFS task, the incorporation of pre-trained word embeddings can always improve the performance of the models, and the incorporation of BERT can further improve the performance.

Again, the probing results do not match the results of RFS on WEBNLG. The effect of incorporating the GloVe embeddings is not significant for c-RNN and ConATT. Although BERT contributes to the learning of DisStat, since most of the entities in WEBNLG are first-mentions, the increased accuracy in DisStat is not sufficient to increase the overall performance of RFS.

On ONTONOTES, pre-trained word embeddings (i.e., GloVe and SGNS) help each model learn significantly more information about almost every feature. Also, consistent with the RFS results, BERT can dramatically improve the models' ability to capture information about all features.

Comparing different RF classifications. Given our expectation $\mathcal{E}_8$, which postulates that more fine-grained classification models would learn more linguistic information, we compare the results of different types of classifications.

On WEBNLG, it seems that the models learn different information by using different sets of labels (classes). For example, 2-way classification (i.e., pronominalisation) helps c-RNN-EL learn more about referential status. However, for models with attention mechanism (i.e., ConATT-EL, ConATT-EL+GloVe and c-RNN-EL+BERT-RT), referential status is better learned in 4-way classification models. Also, in the case of ConATT-EL(+GloVe), we find that more fine-grained classifications help the model learn more about the meta-prominence (i.e., MetaPro).

On ONTONOTES, we found no significant difference between the amount of information learned by the models trained for more fine-grained classifications (3-, 4-, and 5-way classifications). As for 2-way classification, all models except c-RNN-PN+BERT learn less information on ONTONOTES-EN. If we train a model on ONTONOTES-ZH$_{\leq 512}$ for 2-way classification (i.e., whether the target referent is realised as an overt RE or as a ZP), the model learns less information about every feature. This is consistent with our expectation that fine-grained classifications provide more supervision signals for a model to learn more linguistic information than coarse-grained classifications. The reason is that, at least for the RFS task, fine-grained classifications are closer to human behaviour.

6.6. Summary

Based on our probing experiments, each model was able to learn information about referential status, syntactic position, and recency to varying degrees. The models had difficulty acquiring information that required an overall understanding of the entire document or corpus.

We found that the WEBNLG probing results were not able to explain the RFS results, as models that learned more useful information performed worse. In contrast, the probing results on ONTONOTES (which is considered to contain a more realistic use of REs) can explain the RFS results. This suggests that the choice of corpora can influence the interpretability of the findings.

7. General discussion

7.1. Are the results consistent with our expectations?

In what follows, we summarise our findings with respect to our expectations as outlined in Section 3.6.

1. Attention-based models do *not* always outperform the simpler BiGRU-based models. In contrast, the simpler BiGRUs often work better than attention-based models;
2. Pre-trained word embeddings and language models do help with ONTONOTES models, but often *not* with WEBNLG;
3. Models trained on ONTONOTES have lower performance than models trained on WEBNLG;
4. Models using proper names as their referent representations perform better than those using entity labels;
5. Models that work well in English generalise well to Chinese;
6. Chinese RFS models benefit more from using contextual representations than English RFS models, but such a statement is not conclusive (see discussion in Section 5);
7. Models with better RFS performance learn more relevant linguistic information on ONTONOTES but not on WEBNLG;
8. RFS models do *not* always learn more useful information when trained for more fine-grained classifications, but those trained on binary classifications always learn less useful information;
9. Probing results for English and Chinese are similar.

A few of these points are worth discussing in more detail. We will focus on the differences between the two corpora, on the way in which entities are represented, and on the modelling of zero pronouns, as an example of the challenges of modelling a variety of languages. Finally, we will discuss the limitations of our use of probing classifiers.

7.2. WEBNLG *vs.* ONTONOTES

The WEBNLG dataset contains only English-language data, while ONTONOTES contains data in English and Chinese. Furthermore, these two datasets were constructed using different methodologies. Another difference is that almost all referents in the WEBNLG test set also appear in the training set, whereas only a few referents in the ONTONOTES test set appear in its training set. Therefore, it is hard to use the results on the two datasets to conduct controlled comparisons between them. Nevertheless, compared to WEBNLG, the texts in ONTONOTES seem intuitively more natural and the REs are closer to human behaviour. Regarding this intuition, we have the following observations.

First, given the discussion in Section 6, the difficulty of each probing task follows the following order: GloPro $\succ$ DistAnt $\succ$ {SenStat, IntRef} $\succ$ {DisStat, Syn, LocPro}, where A $\succ$ B means that A is harder than B. Theoretically, if a probing task is harder, it is also harder for an RFS model to learn the corresponding task, and the probing classifier, therefore, has lower performance. This theoretical assumption is confirmed when the ONTONOTES dataset is used. For example, since SenStat and IntRef are simpler than DistAnt, each model performs better on either SenStat or IntRef than on DistAnt. Using the WEBNLG dataset, we, unfortunately, did not find a clear correlation between the difficulty of the probing tasks and the performance of a probing classifier.

Second, the aim of the probing study is to understand what and how much linguistic information each model can learn, and to use the results to interpret the models' behaviour. Intuitively, if a model learns more linguistic information than other models, it will achieve better RFS classification performance (i.e., our expectation $\mathcal{E}_3$). However, in WEBNLG, we found that the models that learned more information (according to the probing results) performed worse than those that acquired less linguistic information. One possible explanation is that models that have lower capability to obtain high-level linguistic features are more likely to learn artefacts, and these artefacts may help models perform better. For example, one can imagine a model trained on WEBNLG that has no sense of language and never uses pronouns because 85% of the REs in WEBNLG are first-mentions which are almost never realised as pronouns. Learning this simple "rule" helps the model achieve better performance but know nothing about "language". The situation is different when testing models on ONTONOTES, whose texts and uses of REs are more realistic than WEBNLG. As discussed in Section 6.5, in most cases, the model that performs poorly on probing tasks also does not perform well on RFS classification.

Third, pre-trained word embeddings and language models have proven effective in many NLP tasks. However, in WEBNLG, we found that neither word embeddings (i.e., GloVe) nor pre-trained language models (i.e., BERT) help in RFS classification. Such an abnormal phenomenon is not observed when using the ONTONOTES dataset. The ONTONOTES models that incorporate pre-trained word embeddings and language models almost always (except for 4-way classification on ONTONOTES-EN) perform better than those that do not.

7.3. Entity representations

We have explored two ways of representing entities in RFS: proper name (PN) and entity label (EL). There appears to be a trade-off: EL helps a model identify mentions of the target entity in pre- and post-context, but hinders it in handling unseen entities. In contrast, PN makes it more difficult for a model to identify pre- and post-mentions (especially if the PN consists of multiple words), but helps it to model unseen entities. The experiments have shown that PN is better at representing entities in realistic RE datasets.

However, PN and EL have a common shortcoming: both are unable to represent overlapping REs. Therefore, in creating the ONTONOTES corpus, we used only maximal spans and overlooked the embedded REs. For example, consider the following sentence:

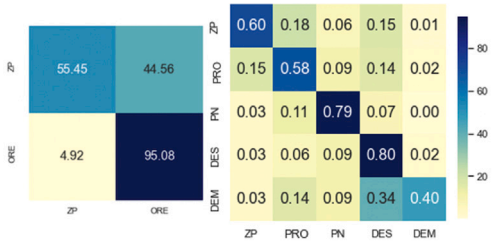

Fig. 4. Confusion Matrix for Chinese 2-way c-RNN$_c$-PN+BERT (left) and Confusion Matrix for Chinese 5-way c-RNN$_c$-PN+BERT (right).

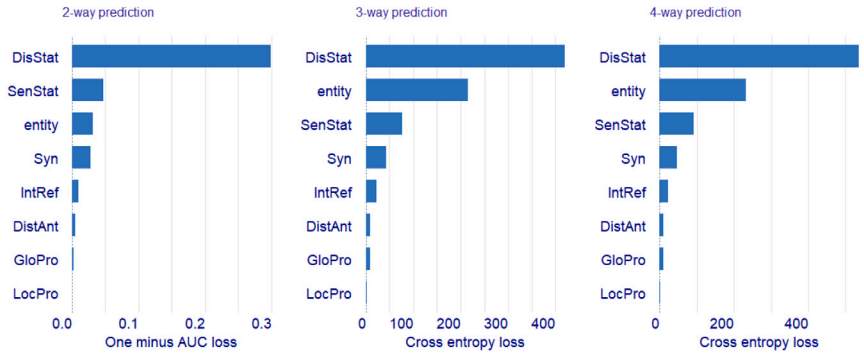

Fig. B.5. Feature importance of the XGBoost classifiers for predictions on ONTONOTES-EN.

(1) He lost his seven children plus his wife and his mom.

In this example, the RE "*his seven children plus his wife and his mom*" contains five REs that refer to five different referents, including "*his seven children plus his wife and his mom*", "*his*", "*his seven children*", "*his wife*", and "*his mom*", but the last four are missing from the corpus due to the limitation of the entity representation methodologies we used.

7.4. Modelling zero pronouns

One of our main reasons for examining Chinese RFS models was to assess whether neural RFS can model ZPs or not. Based on the results of the experiments, we concluded that of all the models tested, c-RNN$_c$-PN+BERT works best. It works remarkably well on using ZPs in a pragmatically natural way. Now, we look more closely at how well it models the use of ZPs.

When we compare the confusion matrices for the 5-way classification and the 2-way classification in Fig. 4, we find that fine-grained supervision helps to better choose between ZPs and overt REs. Let us focus on the 5-way classification to find out which referential form is always confused with ZPs by the model. We observe that the use of ZPs was quite often confused with the use of pronouns by the model. According to linguistic theory, both pronominalisation and pro-drop happen when the target referent is salient enough in the given discourse. Therefore, it is understandable that ZPs and pronouns are easily confused since it is hard for a model to make such a fine-grained decision about when the target referent is salient enough for pronominalisation but not salient enough for pro-drop. Additionally, the use of ZPs is also easily confused with the use of descriptions. One possible explanation is that ONTONOTES is not a balanced dataset. 45% of the REs in the dataset are descriptions, while only 13.6% of them are ZPs. Such an unbalanced distribution causes the trained model to be biased towards non-descriptions (i.e., ZPs, pronouns, proper names, and demonstratives).

In terms of learned linguistic information, we found that in the Chinese 2-way classification (i.e., deciding whether or not to use ZP), the models were good at acquiring information about the syntactic position and referential status. This is consistent with the use of ZPs in ONTONOTES. Specifically, we found that out of 9897 ZPs, 9827 instances are in the subject position, and 8944 instances are discourse-old. This suggests that c-RNN$_c$-PN+BERT does well in modelling human use of ZPs and has not simply learned artefacts from the corpus.

7.5. Limitations of probing classifiers

It is worth noting that probing has its own shortcomings. Firstly, low probing performance does not always mean that the feature is not encoded, but could also mean that such a feature does not matter for RFS. To mitigate this problem, we conducted a complementary ML-based analysis of variable importance. In this analysis, referential status and syntactic position emerged as the factors with the highest contributions. These features were also predicted very well in the probing tasks. However, these results

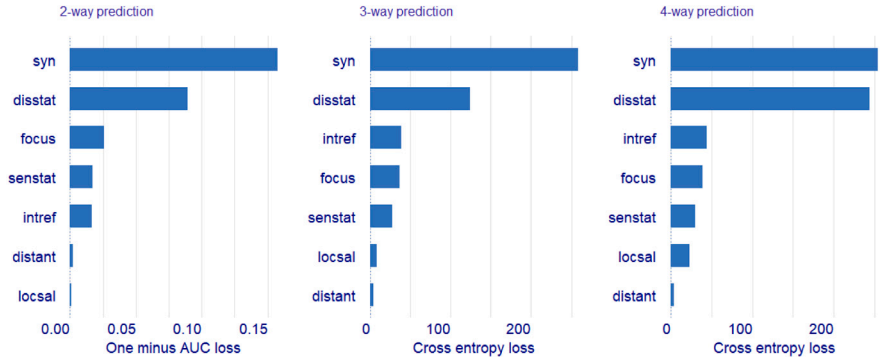

Fig. B.6. Feature importance of the `XGBoost` classifiers for predictions on ONTONOTES-ZH.

Table A.13
Evaluation results of our word-based RFS-PN systems on the ONTONOTES-ZH$_{\leq 512}$ dataset.

Model	5-way			4-way			3-way			2-way		
	P	R	F	P	R	F	P	R	F	P	R	F
c-RNN-PN	51.13	47.14	48.63	54.70	54.02	54.18	57.63	53.79	55.16	66.19	63.22	64.40
+SGNS	53.40	53.33	53.16	57.91	59.12	58.19	60.17	57.49	58.52	70.87	65.22	67.30
ConATT-PN	48.52	45.15	46.26	56.34	49.92	49.26	56.24	55.70	55.94	65.33	64.28	64.75
+SGNS	50.58	47.04	48.31	54.68	51.85	52.62	59.93	55.79	57.32	67.15	65.29	66.11

Table C.14
Accuracy of each probing task on WEBNLG.

Model	Type	DisStat	SenStat	Syn	DistAnt	IntRef	LocPro	GloPro	MetaPro
Random	–	41.83	22.87	48.99	14.90	22.92	49.84	48.02	25.20
majority	–	46.50	31.00	37.99	23.25	31.00	36.01	40.65	10.97
c-RNN	4-way	84.06	73.72	85.34	53.84	55.43	82.92	56.00	42.32
	3-way	83.72	72.60	83.60	54.78	53.21	81.67	56.70	41.79
	2-way	88.04	73.84	84.00	54.93	52.31	85.69	59.98	41.65
c-RNN +GloVe	4-way	84.85	74.59	87.04	55.67	55.93	83.20	53.53	41.71
	3-way	83.89	67.24	82.48	50.94	51.17	81.44	52.49	42.34
	2-way	88.02	71.25	82.67	53.67	51.43	85.22	63.17	41.03
c-RNN +BERT	4-way	90.64	78.04	82.71	56.91	54.30	81.67	54.24	43.07
	3-way	84.80	72.29	84.08	54.21	53.25	82.53	57.31	42.80
	2-way	87.28	69.69	84.74	54.19	54.88	82.77	63.07	40.75
ConATT	4-way	87.81	77.11	88.00	57.09	55.88	86.34	60.15	46.14
	3-way	84.39	74.19	86.66	55.26	54.09	84.56	60.61	47.47
	2-way	84.20	73.18	88.44	53.98	53.64	86.75	56.39	41.81
ConATT +GloVe	4-way	87.82	77.70	87.24	57.52	55.22	85.69	58.54	49.94
	3-way	84.35	72.83	88.91	54.23	51.96	86.80	59.05	46.36
	2-way	84.38	73.21	86.96	56.14	53.33	85.27	62.46	39.63

should be taken with a pinch of salt: the variable importance was conducted with the ML model and not with the neural models. We cannot be certain that the same features contribute to all models similarly: a feature could be very important in the machine learning model but not in the neural models.

Furthermore, some researchers have questioned the validity of probing methods. They have found that it is difficult for a probing classifier to distinguish between "learning the probing task" and "extracting the encoded linguistic information" (Hewitt and Liang, 2019; Kunz and Kuhlmann, 2020). This suggests that higher performance of a probing classifier does not necessarily mean that more linguistic information was encoded. This prevents us from directly quantifying *how well* linguistic information was learned based on the performance of probing classifiers and requires us to draw our conclusions more carefully. In future, we plan to test other model explanation techniques, e.g., probing classifiers with control tasks (Hewitt and Liang, 2019) and attention analysis (Bibal et al., 2022).

7.6. Concluding thoughts

One interpretation of Searle's "Chinese Room" experiment (Searle, 1980) is that an NLP algorithm may display human-like behaviour without understanding much about human language. Neural NLP systems have sometimes been seen as a case in point

Table C.15
Accuracy of our baselines as well as RFS-PN models on each probing task on the ONTONOTES-EN dataset.

Model	Type	DisStat	SenStat	Syn	DistAnt	IntRef	LocPro	GloPro
Random	–	49.77	32.27	50.10	23.75	32.40	48.21	49.53
majority	–	35.88	20.39	33.39	15.29	20.39	40.50	38.68
c-RNN-PN	4-way	63.39	50.76	74.67	36.31	44.81	67.86	50.32
	3-way	63.30	50.45	75.55	36.78	42.83	68.26	49.71
	2-way	63.31	49.72	73.82	35.75	43.03	65.72	45.76
c-RNN-PN +GloVe	4-way	64.24	51.39	76.75	37.09	44.94	67.26	51.44
	3-way	64.44	52.69	78.06	37.55	45.89	70.66	53.28
	2-way	64.26	49.34	75.06	35.87	45.04	67.22	47.49
c-RNN-PN +BERT	4-way	86.00	72.17	79.83	66.53	69.85	82.32	68.47
	3-way	83.74	71.56	81.17	65.35	68.03	85.05	67.82
	2-way	81.82	69.33	78.05	63.46	65.11	81.85	66.35
ConATT-PN	4-way	62.95	46.63	73.34	33.33	43.52	66.30	48.89
	3-way	61.87	45.92	74.76	31.91	41.64	67.51	48.61
	2-way	59.46	41.73	63.72	30.18	40.85	59.96	47.51
ConATT-PN +GloVe	4-way	63.41	50.49	79.95	36.03	43.17	70.27	49.86
	3-way	61.79	45.39	79.00	33.03	41.53	68.46	48.97
	2-way	61.56	44.35	73.97	31.53	42.81	63.31	48.39

Table C.16
Accuracy of RFS-PN models on each probing task on the ONTONOTES-ZH$_{\leq 512}$ dataset.

Model	Type	DisStat	SenStat	Syn	DistAnt	IntRef	LocPro	GloPro
Random	–	49.93	32.70	49.79	23.81	33.01	46.44	44.27
majority	–	36.43	19.95	36.62	14.96	19.95	43.27	45.09
c-RNN$_c$-PN	5-way	62.80	45.89	75.94	28.49	45.78	65.54	52.03
	4-way	61.80	43.39	74.74	27.73	44.65	63.44	46.64
	3-way	61.19	41.52	76.11	26.43	41.13	61.70	45.76
	2-way	58.06	36.30	76.96	24.11	36.49	58.82	45.54
c-RNN$_c$-PN +SGNS	5-way	63.52	47.24	77.28	30.71	46.13	66.11	50.37
	4-way	62.90	46.96	77.18	30.78	47.81	66.16	48.20
	3-way	62.87	42.54	77.81	27.51	43.59	64.17	46.11
	2-way	60.45	38.24	77.12	24.27	37.61	64.09	46.12
c-RNN$_c$-PN +BERT	5-way	75.20	57.07	78.68	39.54	57.69	70.93	55.17
	4-way	73.96	57.66	78.15	37.12	56.90	69.68	46.60
	3-way	73.77	56.29	79.67	35.77	55.96	73.24	45.59
	2-way	68.10	52.08	79.84	29.71	52.36	71.30	45.07
ConATT$_c$-PN	5-way	62.33	43.17	73.94	28.92	45.05	63.99	47,16
	4-way	61.91	43.15	67.48	26.41	44.15	57.31	47.27
	3-way	59.54	39.55	68.78	24.47	39.13	55.27	45.73
	2-way	52.10	32.85	65.67	21.78	32.66	49.38	45.35
ConATT$_c$-PN +SGNS	5-way	63.48	45.60	77.27	29.77	46.84	64.16	50.72
	4-way	61.97	44.63	74.65	28.19	46.61	64.49	47.27
	3-way	58.79	38.78	74.09	24.66	38.51	60.19	45.73
	2-way	60.09	39.53	72.90	22.13	34.88	61.43	45.35

because, allegedly, they do not embody any insights into the questions that linguists are interested in. Our probing experiments of Section 6 show that, for the neural RFS models discussed in this paper, this assessment would not be fair, because these models were shown to have learned key linguistic concepts such as referential status, syntactic position, recency, and prominence.

Another lesson from our investigations is that researchers in REG, and probably elsewhere in NLG and NLP as well, would sometimes be wise to reflect on the limits of the validity of their findings.

For evidently, findings about one language cannot always be generalised to other languages, even if these findings are on a relatively abstract level; we have seen this when we compared English with Chinese referential behaviour. Perhaps more worryingly, findings about one corpus (in a given language) cannot always be generalised to other corpora (even in that same language). A corpus is a data sample, and every data sample gives rise to the question of what that sample is representative of. We found that, in regard of REG use, the WEBNLG corpus is limited to very short texts that do not offer a playing field on which pronouns can play the kind of role that they play in longer texts. Although such problems with generalisation across datasets might seem obvious from a perspective of the natural and social sciences – and have been occasionally discussed in corpus linguistics as well, see e.g. Biber (1993), Sinclair (2005), Moreno Fernández (2004) – it appears to sometimes be overlooked in some areas of modern, data intensive NLP, when researchers fail to say what type of language use their corpora are thought to be representative of.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgements

Fahime Same is funded by the German Research Foundation (DFG)– Project-ID 281511265– SFB 1252 "Prominence in Language".

Appendix A. Complementary results

Table A.13 shows the results of our word-based RFS-PN models on the ONTONOTES-ZH$_{\leq 512}$ dataset.

Appendix B. Importance analysis on ONTONOTES

Fig. B.5 and B.6 show the importance analysis on the ONTONOTES-EN and ONTONOTES-ZH datasets respectively.

Appendix C. Accuracy of probing experiments

Tables C.14, C.15, and C.16 report the accuracy of each probing task on each dataset.

References

Alt, C., Gabryszak, A., Hennig, L., 2020. Probing linguistic features of sentence-level representations in neural relation extraction. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, Online, pp. 1534–1545. http://dx.doi.org/10.18653/v1/2020.acl-main.140, URL https://aclanthology.org/2020.acl-main.140.

Ariel, M., 1990. Accessing Noun-Phrase Antecedents. Routledge.

Arnold, J.E., 2001. The effect of thematic roles on pronoun use and frequency of reference continuation. Discourse Process. 31 (2), 137–162.

Arnold, J.E., 2010. How speakers refer: The role of accessibility. Lang. Linguist. Compass 4 (4), 187–203.

Arnold, J.E., Griffin, Z.M., 2007. The effect of additional characters on choice of referring expression: Everyone counts. J. Mem. Lang. 56 (4), 521–536.

Barredo Arrieta, A., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., Garcia, S., Gil-Lopez, S., Molina, D., Benjamins, R., Chatila, R., Herrera, F., 2020. Explainable artificial intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. Inf. Fusion 58, 82–115. http://dx.doi.org/10.1016/j.inffus.2019.12.012, URL https://www.sciencedirect.com/science/article/pii/S1566253519308103.

Belinkov, Y., Durrani, N., Dalvi, F., Sajjad, H., Glass, J., 2017. What do neural machine translation models learn about morphology? In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Association for Computational Linguistics, Vancouver, Canada, pp. 861–872. http://dx.doi.org/10.18653/v1/P17-1080, URL https://aclanthology.org/P17-1080.

Belz, A., Kow, E., Viethen, J., Gatt, A., 2010. Generating referring expressions in context: The GREC task evaluation challenges. In: Krahmer, E., Theune, M. (Eds.), Empirical Methods in Natural Language Generation: Data-Oriented Methods and Empirical Evaluation. In: Lecture Notes in Computer Science, Vol. 5790, Springer, pp. 294–327. http://dx.doi.org/10.1007/978-3-642-15573-4_15.

Belz, A., Varges, S., 2007. Generation of repeated references to discourse entities. In: Proceedings of the Eleventh European Workshop on Natural Language Generation. Association for Computational Linguistics, pp. 9–16.

Bibal, A., Cardon, R., Alfter, D., Wilkens, R., Wang, X., François, T., Watrin, P., 2022. Is attention explanation? An introduction to the debate. In: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Association for Computational Linguistics, Dublin, Ireland, pp. 3889–3900. http://dx.doi.org/10.18653/v1/2022.acl-long.269, URL https://aclanthology.org/2022.acl-long.269.

Biber, D., 1993. Representativeness in corpus design. Lit. Linguist. Comput. 8 (4), 243–257.

Biecek, P., Burzykowski, T., 2021. Explanatory Model Analysis: Explore, Explain, and Examine Predictive Models. CRC Press.

Brennan, S.E., 1995. Centering attention in discourse. Lang. Cognit. Process. 10 (2), 137–167.

Cao, M., Cheung, J.C.K., 2019. Referring expression generation using entity profiles. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). Association for Computational Linguistics, Hong Kong, China, pp. 3163–3172. http://dx.doi.org/10.18653/v1/D19-1312, URL https://aclanthology.org/D19-1312.

Castelvecchi, D., 2016. Can we open the black box of AI? Nat. News 538 (7623), 20.

Castro Ferreira, T., Krahmer, E., Wubben, S., 2016. Towards more variation in text generation: Developing and evaluating variation models for choice of referential form. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Association for Computational Linguistics, Berlin, Germany, pp. 568–577. http://dx.doi.org/10.18653/v1/P16-1054, URL https://aclanthology.org/P16-1054.

Castro Ferreira, T., Moussallem, D., Kádár, Á., Wubben, S., Krahmer, E., 2018a. NeuralREG: An end-to-end approach to referring expression generation. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Association for Computational Linguistics, Melbourne, Australia, pp. 1959–1969. http://dx.doi.org/10.18653/v1/P18-1182, URL https://aclanthology.org/P18-1182.

Castro Ferreira, T., Moussallem, D., Krahmer, E., Wubben, S., 2018b. Enriching the webNLG corpus. In: Proceedings of the 11th International Conference on Natural Language Generation. Association for Computational Linguistics, Tilburg University, The Netherlands, pp. 171–176. http://dx.doi.org/10.18653/v1/W18-6521, URL https://aclanthology.org/W18-6521.

Chafe, W., 1976. Givenness, contrastiveness, definiteness, subjects, topics, and point of view. Subj. Top..

Chen, G., 2022. Computational Generation of Chinese Noun Phrases (Ph.D. thesis). Utrecht University.

Chen, G., van Deemter, K., 2020. Lessons from computational modelling of reference production in mandarin and english. In: Proceedings of the 13th International Conference on Natural Language Generation. Association for Computational Linguistics, Dublin, Ireland, pp. 263–272, URL https://aclanthology.org/2020.inlg-1.33.

Chen, G., van Deemter, K., 2022. Understanding the use of quantifiers in Mandarin. In: Findings of the Association for Computational Linguistics: AACL-IJCNLP 2022. Association for Computational Linguistics, Online only, pp. 73–80, https://aclanthology.org/2022.findings-aacl.7.

Chen, G., van Deemter, K., Lin, C., 2018. Modelling pro-drop with the rational speech acts model. In: Proceedings of the 11th International Conference on Natural Language Generation. Association for Computational Linguistics, Tilburg University, The Netherlands, pp. 159–164. http://dx.doi.org/10.18653/v1/W18-6519, URL https://aclanthology.org/W18-6519.

Chen, T., Guestrin, C., 2016. Xgboost: A scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. KDD '16, Association for Computing Machinery, New York, NY, USA, pp. 785–794. http://dx.doi.org/10.1145/2939672.2939785, URL https://doi.org/10.1145/2939672.2939785.

Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y., 2014. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). Association for Computational Linguistics, Doha, Qatar, pp. 1724–1734. http://dx.doi.org/10.3115/v1/D14-1179, URL https://aclanthology.org/D14-1179.

Cunha, R., Castro Ferreira, T., Pagano, A., Alves, F., 2020. Referring to what you know and do not know: Making referring expression generation models generalize to unseen entities. In: Proceedings of the 28th International Conference on Computational Linguistics. International Committee on Computational Linguistics, Barcelona, Spain (Online), pp. 2261–2272. http://dx.doi.org/10.18653/v1/2020.coling-main.205, URL https://aclanthology.org/2020.coling-main.205.

Devlin, J., Chang, M.-W., Lee, K., Toutanova, K., 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. 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). Association for Computational Linguistics, Minneapolis, Minnesota, pp. 4171–4186. http://dx.doi.org/10.18653/v1/N19-1423, URL https://aclanthology.org/N19-1423.

Dosilovic, F.K., Brcic, M., Hlupic, N., 2018. Explainable artificial intelligence: A survey. In: 2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO). IEEE, http://dx.doi.org/10.23919/mipro.2018.8400040.

Fang, R., Doering, M., Chai, J.Y., 2015. Embodied collaborative referring expression generation in situated human-robot interaction. In: Proceedings of the Tenth Annual ACM/IEEE International Conference on Human-Robot Interaction. pp. 271–278.

Fukumura, K., van Gompel, R.P., 2011. The effect of animacy on the choice of referring expression. Lang. Cognit. Process. 26 (10), 1472–1504.

Gardent, C., Shimorina, A., Narayan, S., Perez-Beltrachini, L., 2017. Creating training corpora for NLG micro-planners. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Association for Computational Linguistics, Vancouver, Canada, pp. 179–188. http://dx.doi.org/10.18653/v1/P17-1017, URL https://aclanthology.org/P17-1017.

Giulianelli, M., Harding, J., Mohnert, F., Hupkes, D., Zuidema, W., 2018. Under the hood: Using diagnostic classifiers to investigate and improve how language models track agreement information. In: Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP. Association for Computational Linguistics, Brussels, Belgium, pp. 240–248. http://dx.doi.org/10.18653/v1/W18-5426, URL https://aclanthology.org/W18-5426.

Greenbacker, C., McCoy, K., 2009. UDel: generating referring expressions guided by psycholinguistic findings. In: Proceedings of the 2009 Workshop on Language Generation and Summarisation. Association for Computational Linguistics, pp. 101–102.

Grosz, B.J., Joshi, A.K., Weinstein, S., 1995. Centering: A framework for modeling the local coherence of discourse. Comput. Linguist. 21 (2), 203–225, URL https://aclanthology.org/J95-2003.

Gundel, J.K., Hedberg, N., Zacharski, R., 1993. Cognitive status and the form of referring expressions in discourse. Language 274–307.

Hendrickx, I., Daelemans, W., Luyckx, K., Morante, R., Van Asch, V., 2008. CNTS: Memory-based learning of generating repeated references. In: Proceedings of the Fifth International Natural Language Generation Conference. Association for Computational Linguistics, Salt Fork, Ohio, USA, pp. 194–195, URL https://aclanthology.org/W08-1129.

Henschel, R., Cheng, H., Poesio, M., 2000. Pronominalization revisited. In: Proceedings of the 18th Conference on Computational Linguistics-Volume 1. Association for Computational Linguistics, pp. 306–312.

Hewitt, J., Liang, P., 2019. Designing and interpreting probes with control tasks. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). Association for Computational Linguistics, Hong Kong, China, pp. 2733–2743. http://dx.doi.org/10.18653/v1/D19-1275, URL https://aclanthology.org/D19-1275.

Hinterwimmer, S., 2019. Prominent protagonists. J. Pragmat. 154, 79–91.

Huang, C.-T.J., 1984. On the distribution and reference of empty pronouns. Linguistic Inquiry 531–574.

Kehler, A., Kertz, L., Rohde, H., Elman, J.L., 2008. Coherence and coreference revisited. J. Semant. 25 (1), 1–44.

Kibrik, A.A., Khudyakova, M.V., Dobrov, G.B., Linnik, A., Zalmanov, D.A., 2016. Referential choice: Predictability and its limits. Front. Psychol. 7, 1429. http://dx.doi.org/10.3389/fpsyg.2016.01429, URL https://www.frontiersin.org/article/10.3389/fpsyg.2016.01429.

Krahmer, E., van Deemter, K., 2012. Computational generation of referring expressions: A survey. Comput. Linguist. 38 (1), 173–218.

Kunz, J., Kuhlmann, M., 2020. Classifier probes may just learn from linear context features. In: Proceedings of the 28th International Conference on Computational Linguistics. International Committee on Computational Linguistics, Barcelona, Spain (Online), pp. 5136–5146. http://dx.doi.org/10.18653/v1/2020.coling-main.450, URL https://aclanthology.org/2020.coling-main.450.

Li, S., Zhao, Z., Hu, R., Li, W., Liu, T., Du, X., 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). Association for Computational Linguistics, Melbourne, Australia, pp. 138–143. http://dx.doi.org/10.18653/v1/P18-2023, URL https://aclanthology.org/P18-2023.

Mao, J., Huang, J., Toshev, A., Camburu, O., Yuille, A.L., Murphy, K., 2016. Generation and comprehension of unambiguous object descriptions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 11–20.

Moreno Fernández, F., 2004. Corpora of spoken spanish language-the representativeness issue. In: The First International Conference on Linguistic Informatics. State of the Art and the Future. Tokyo University of Foreign Studies Tokio, pp. 49–76.

Newnham, R., 1971. About Chinese. Penguin Books Ltd.

Pandit, O., Hou, Y., 2021. Probing for bridging inference in transformer language models. In: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Association for Computational Linguistics, Online, pp. 4153–4163, URL https://www.aclweb.org/anthology/2021.naacl-main.327.

Pennington, J., Socher, R., Manning, C., 2014. Glove: Global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). Association for Computational Linguistics, Doha, Qatar, pp. 1532–1543. http://dx.doi.org/10.3115/v1/D14-1162, URL https://aclanthology.org/D14-1162.

Pradhan, S., Moschitti, A., Xue, N., Uryupina, O., Zhang, Y., 2012. CoNLL-2012 shared task: Modeling multilingual unrestricted coreference in OntoNotes. In: Joint Conference on EMNLP and CoNLL - Shared Task. Association for Computational Linguistics, Jeju Island, Korea, pp. 1–40, URL https://aclanthology.org/W12-4501.

Prasad, R., 2003. Constraints on the* Generation of Referring Expressions, with Special Reference To Hindi. University of Pennsylvania.

Prince, E.F., 1981. Towards a taxonomy of given-new information. Radic. Pragmat..

Reiter, E., 2017. A commercial perspective on reference. In: Proceedings of the 10th International Conference on Natural Language Generation. Association for Computational Linguistics, Santiago de Compostela, Spain, pp. 134–138. http://dx.doi.org/10.18653/v1/W17-3519, URL https://aclanthology.org/W17-3519.

Reiter, E., Dale, R., 2000. Building Natural Language Generation Systems. In: Studies in Natural Language Processing, Cambridge University Press, http://dx.doi.org/10.1017/CBO9780511519857.

Same, F., Chen, G., Van Deemter, K., 2022. 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). Association for Computational Linguistics, Dublin, Ireland, pp. 5554–5567. http://dx.doi.org/10.18653/v1/2022.acl-long.380, URL https://aclanthology.org/2022.acl-long.380.

Same, F., van Deemter, K., 2020. A linguistic perspective on reference: Choosing a feature set for generating referring expressions in context. In: Proceedings of the 28th International Conference on Computational Linguistics. International Committee on Computational Linguistics, Barcelona, Spain (Online), pp. 4575–4586. http://dx.doi.org/10.18653/v1/2020.coling-main.403, URL https://aclanthology.org/2020.coling-main.403.

Searle, J.R., 1980. Minds, brains, and programs. Behav. Brain Sci. 3 (3), 417–424.

Siddharthan, A., Nenkova, A., McKeown, K., 2011. Information status distinctions and referring expressions: An empirical study of references to people in news summaries. Comput. Linguist. 37 (4), 811–842.

Sinclair, J., 2005. Corpus and text-basic principles. In: Developing Linguistic Corpora: A Guide To Good Practice. Vol. 92, pp. 1–16.

Sorodoc, I.-T., Gulordava, K., Boleda, G., 2020. Probing for referential information in language models. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, Online, pp. 4177–4189. http://dx.doi.org/10.18653/v1/2020.acl-main.384, URL https://aclanthology.org/2020.acl-main.384.

Sun, R., 2008. The Cambridge Handbook of Computational Psychology. Cambridge University Press.

Sutskever, I., Vinyals, O., Le, Q.V., 2014. Sequence to sequence learning with neural networks. In: Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N., Weinberger, K.Q. (Eds.), Advances in Neural Information Processing Systems. Vol. 27, Curran Associates, Inc., URL https://proceedings.neurips.cc/paper/2014/file/a14ac55a4f27472c5d894ec1c3c743d2-Paper.pdf.

Torroba Hennigen, L., Williams, A., Cotterell, R., 2020. Intrinsic probing through dimension selection. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). Association for Computational Linguistics, Online, pp. 197–216. http://dx.doi.org/10.18653/v1/2020.emnlp-main.15, URL https://aclanthology.org/2020.emnlp-main.15.

van Deemter, K., 2016. Computational Models of Referring: A Study in Cognitive Science. MIT Press.

Vicente, K.J., Wang, J.H., 1998. An ecological theory of expertise effects in memory recall. Psychol. Rev. 105 (1), 33.

von Heusinger, K., Schumacher, P.B., 2019. Discourse prominence: Definition and application. J. Pragmat. 154, 117–127.

Walker, M., Cote, S., Iida, M., 1994. Japanese discourse and the process of centering. Comput. Linguist. 20 (2), 193–232.

West, D., 2018. The Future of Work : Robots, AI, and Automation. Brookings Institution Press.

Yang, C.L., Gordon, P.C., Hendrick, R., Wu, J.T., 1999. Comprehension of referring expressions in Chinese. Lang. Cognit. Process. 14 (5–6), 715–743.

Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., Hovy, E., 2016. Hierarchical attention networks for document classification. In: Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Association for Computational Linguistics, San Diego, California, pp. 1480–1489. http://dx.doi.org/10.18653/v1/N16-1174, URL https://aclanthology.org/N16-1174.