Advances in Natural Language Generation

Castro Ferreira, Thiago

Publication date:
2018

Document Version
Publisher's PDF, also known as Version of record

Advances in Natural Language Generation

Generating Varied Outputs from Semantic Inputs

Thiago Castro Ferreira

Advances in Natural Language Generation
Generating Varied Outputs from Semantic Inputs

Thiago Castro Ferreira
PhD Thesis
Tilburg University, 2018

TiCC PhD Series No. 64

Financial Support was received from the National Council of Scientific and Technological Development from Brazil (CNPq), Grant 203065/2014-0.

Cover design: Ariadni Blom
Design: Thiago Castro Ferreira
Print: Ipskamp Printing

Advances in Natural Language Generation

Generating Varied Outputs from Semantic Inputs

PROEFSCHRIFT

ter verkrijging van de graad van doctor
aan Tilburg University
op gezag van de rector magnificus,
prof.dr. E.H.L. Aarts,
in het openbaar te verdedigen ten overstaan van
een door het college voor promoties aangewezen commissie
in de aula de Universiteit
op woensdag 19 september 2018 om 14.00 uur

door
Thiago Castro Ferreira
geboren op 5 februari 1990 te São Paulo, Brazilië

Promotores
Prof. Dr. E. Krahmer
Dr. S. Wubben

Commissieleden
Prof. Dr. A. Gatt
Prof. Dr. A. van den Bosch
Dr. C. Gardent
Dr. F. Schilder
Dr. M.B. Goudbeek

"The idealist is incorrigible: if he is thrown out of his heaven he makes an ideal of his hell."

– Friedrich Nietzsche

Contents

1 Introduction

Natural Language Generation (NLG) – also known as Automatic Text Generation – is the computational process of generating understandable natural language text from non-linguistic input data (Reiter & Dale, 2000; Gatt & Krahmer, 2018). The interest in NLG systems has increased in recent years, in part because practical applications of these techniques are becoming increasingly feasible. These include, for example, applications in weather forecasts (Goldberg et al., 1994; Sripada et al., 2004; Belz, 2008; Konstas & Lapata, 2013) and neonatal intensive care reports for doctors and caregivers (Reiter, 2007; Portet et al., 2009) as well as news reports written by "robot-journalists" (Clerwall, 2014).

While most current NLG systems are capable of generating informative and grammatically correct texts, evaluation studies also reveal that these system-generated texts are rated differently from human-produced ones (Stent et al., 2005; Belz & Reiter, 2006; Novikova et al., 2017a).

For example, in a study comparing news articles generated by a robot-journalist with articles produced by human journalists, Clerwall (2014) noticed that readers rated the former as more *informative*, *accurate*, *trustworthy* and *objective* than the latter. However, they also thought that robot-written text was *less pleasant to read*, and even worse, considered it more *boring* than human-generated news.

We believe that this partly negative assessment of robot-written texts might be due to the deterministic nature of the NLG approach, resulting in a lack of variation in the generated outputs. Indeed, many NLG systems produce the same output for a given input, resulting in rather rigid output texts, which can be unpleasant and boring to read, especially when confronted with multiple texts in succession. Human authors, by contrast, can easily express the same communicative idea using a variety of words and phrases, and hence have no problems whatsoever in writing varied texts.

The first problem we address in this thesis is how to take linguistic variation into account during automatic text generation. We approach this problem by zooming in on a core task of NLG: the generation of references to discourse entities, a process commonly known as Referring Expression Generation (REG) (Krahmer & van Deemter, 2012). Initially, we look into traditional REG approaches, in which a model is first used to choose the form of a reference followed by another model which decides on the content of the referring expression given the chosen form. In this particular approach, based on analyses of how human authors generate references in different contexts and settings, we develop data-driven models that introduce linguistic variation in two tasks of REG: (1) the choice of referential form, determining whether a reference should, for instance, take the form of a proper name (“Gal Costa”), a description (“the Brazilian singer”) or a pronoun (“she”); and (2) the generation of proper names,

determining, for example, whether *Gal Costa* should be referred to by her full name ("Maria da Graça Costa Penna Burgos"), first name ("Gal") or last name ("Costa"), once the system has decided a proper name reference should be generated. Next, and in contrast to the the traditional modular style of performing REG, we introduce a novel end-to-end approach which produces varied referring expressions in discourse, simultaneously deciding on form and content. By introducing linguistic variation in both REG designs, we aim to increase the *humanlikeness* of NLG outputs and thereby hopefully improve the appreciation of the readers of generated texts.

The second issue we address in this thesis is not related to the output of NLG systems, but instead to their *input*. While there is a broad consensus among scholars on the output of their systems (i.e., text; potentially in spoken form, Theune et al. 2001; Ferres et al. 2006), there is far less agreement on what the input representations of NLG systems should be. Over the years, a wide range of input formats have been used, including, for example, images (Xu et al., 2015), numeric data (Gkatzia et al., 2014) and semantic representations (Theune et al., 2001). Even for the latter, which is the most popular representation among data-driven models, there is no agreement upon format. Part of the difficulty is that there is a complicated trade-off between the level of specification of the input meaning representation and the complexity of the generation process. Some early NLG systems used highly detailed input representations, thereby limiting the general applicability of these approaches. Other NLG models generated text from less complex input representations that were not so closely associated to the intended linguistic output. By focusing more on the semantics, these approaches generally needed to rely on more complex approaches for converting meaning into text. In general, the more decisions

can be taken within the generation process itself, the more general and abstract the input representations can be (Gatt & Krahmer, 2018).

In this thesis, we explore generation from two distinct and increasingly popular meaning representations, namely Abstract Meaning Representation (Banarescu et al., 2013) and RDF Triples from the Semantic Web (Bizer et al., 2009). For both representations, we develop data-driven NLG models which are based either on traditional NLG approaches (Reiter & Dale, 2000), or on more recent approaches importing concepts from Statistical (Koehn et al., 2003) and Neural (Bahdanau et al., 2015) Machine Translation. A comparison between these different models is a secondary goal of this part of the thesis.

Taken together, the two strands of research in this thesis allows us to generate varied texts from semantic inputs. In the remained of this chapter, we sketch the NLG process in more detail, followed by a further introduction to the studies in this thesis.

1.1 Natural Language Generation

The first NLG systems date from the 60s of the 20th century (Yngve, 1961; Friedman, 1969). Their goal was to evaluate the adequacy of grammars by generating sentences without a communicative goal. Yngve (1961), for instance, pointed out that the sentences generated by his generative grammar "were for the most part quite grammatical, though of course nonsensical". Subsequent systems filled that gap by generating text from some meaning representation input, such as semantic networks (Simmons & Slocum, 1972).

Up to that moment, NLG systems had focused on *how* to generate text from a pre-determined communicative goal. Later studies started to

emphasize the selection of *what* to say as part of the natural language generation process (McKeown, 1982). Despite some exceptions, of which Appelt (1980) is an early example, the division of the NLG process in these two broad sequential steps is still often applied in the field. First, an NLG system should choose *what* to say, before deciding on *how* to textually realize the selected information, i.e., make the most appropriate syntactic and lexical choices to convert the selected content into a grammatical and coherent text. From now on, we will refer to these two steps as *Content Selection* and *Surface Realization*, respectively.

Many NLG systems are modular, dividing the Content Selection and Surface Realization processes into several further steps, organized both sequentially and hierarchically. Practical application of modular NLG systems can be found in a wide range of domains, such as in sportscasting (Theune et al., 2001; van der Lee et al., 2017), weather (Goldberg et al., 1994; Sripada et al., 2004) and pollen forecast (Turner et al., 2006), safety-oriented summaries of scuba dives (Sripada & Gao, 2007), gas turbine event descriptions (Yu et al., 2007), stock market information (Kukich, 1983), personalized smoking cessation letters (Reiter et al., 2003), neonatal intensive care reports (Reiter, 2007; Portet et al., 2009), encyclopedic data (Duma & Klein, 2013; Androutsopoulos et al., 2013) and many others (McKeown, 1982; Iordanskaja et al., 1992). Most of these systems are rule-based, and in many of them, a pipeline architecture is implemented, along the lines sketched by Reiter & Dale (2000), who argue this architecture is the "consensus".

Recently, more data-driven approaches have started to become popular in NLG. Usually, these approaches make use of parallel corpora where a statistical or machine learning model is trained to map the non-linguistic source side into the natural language target side. Data-driven approaches

have been proposed both for particular tasks of the modular systems as well as to perform Content Selection and Surface Realization in a less modular style towards an end-to-end approach.

Below, we first describe the pipeline architecture (Reiter & Dale, 2000), the most popular among modular models, and discuss data-driven NLG approaches in more detail as well. This will allow us to briefly introduce the core tasks of any NLG system and sketch some of the approaches to these tasks proposed in the field.

1.1.1 Pipeline architecture

According to Reiter & Dale (2000), the pipeline architecture splits Content Selection and Surface Realization in three broad sequential steps: Document Planning, Sentence Planning and Textual Realization. To illustrate each step we provide an example of a (fictitious) NLG system generating summaries of Brazilian female singers. Table 1.1 depicts part of the non-linguistic domain information which could serve as input for this process. In our example, we will show step-by-step how to generate a summary for *Elis_Regina*.

Document Planning

As the name suggests, Document Planning, sometimes also called Macroplanning, concerns the decisions to be taken on the document level. Reiter & Dale (2000) breaks this task up into two subtasks: Content Determination and Text Structuring.

Content Determination is the process of choosing the communicative goals to be realized, whereas Text Structuring is the process responsible for ordering the selected communicative goals in sentences and paragraphs. The approach chosen for Document Planning is generally strongly

Singer	Genre	Influences	Instruments
Elis_Regina	Bossa_Nova MPB Jazz	Billie_Holiday Carmen_Miranda Nancy_Wilson	-
Gal_Costa	Bossa_Nova MPB Samba	Elis_Regina Janis_Joplin Nina_Simone	Guitar Piano
Marisa_Monte	MPB Samba	Carmen_Miranda Elizete_Cardoso Gal_Costa Maria_Bethania	Guitar Drums Piano Ukulele

Table 1.1: Brazilian female singers, together with information on the genre in which they perform, their influences and the instruments they play.

Paragraph	Subject	Predicate	Object
1	Elis_Regina	genre	Bossa_Nova Popular Jazz
1	Elis_Regina	influence	Billie_Holiday Carmen_Miranda Nancy_Wilson

Table 1.2: A document plan for a summary about *Elis_Regina*.

influenced by the task and domain of application; few generic approaches exist (e.g, Barzilay & Lee, 2004; Barzilay & Lapata, 2005). For the sake of illustration, let us assume that the module came up with the document plan (sometimes also called text plan) depicted in Table 1.2, in the format of *subject-predicate-object* triples for a summary about *Elis_Regina*.

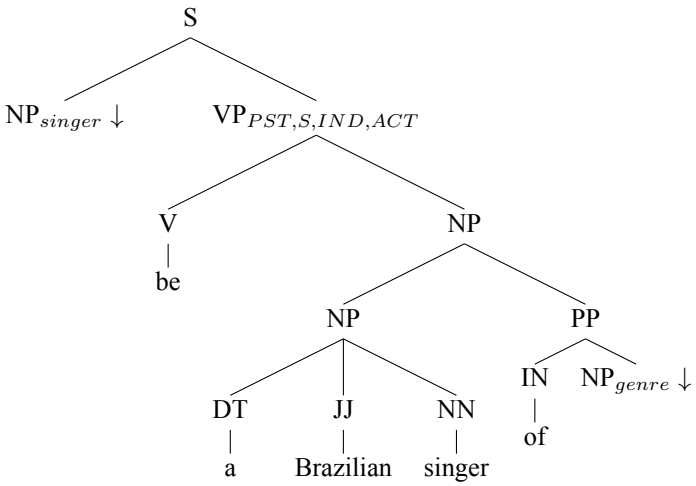
Sentence Planning

Sentence Planning, sometimes also known as Microplanning, is the task of making decisions at the level of a sentence. It receives as input a document plan and performs three tasks in order to generate sentences (Reiter & Dale, 2000): Lexicalization, Aggregation and Referring Expression Generation.

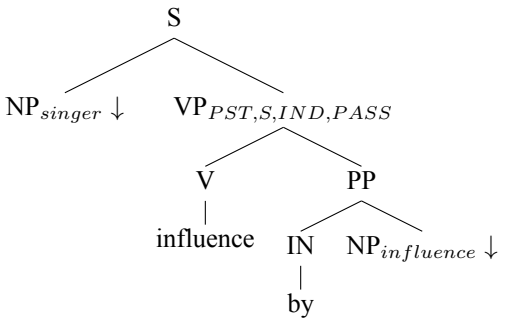
Lexicalization involves finding the proper phrases and words to express the content to be included in each sentence (e.g., Reiter et al., 2005; Smiley et al., 2016). Figure 1.1 depicts syntactic trees which may represent the predicates *genre* and *influence* of our example.

Aggregation is the process in which two or more clauses are merged into a single sentence in order to improve the conciseness and readability of the produced text. Figure 1.2 shows the result of merging the clauses for *genre* and *influence* into a single sentence.

Finally, the 'gaps' in the structure need to be filled with references to the discourse entities. This is typically handled by a Referring Expression Generation (REG) module. REG is one of the tasks that has received most scholarly attention within NLG (Krahmer & van Deemter, 2012). According to Reiter & Dale (2000), the first decision to be taken in REG is the choice of referential form, meaning whether a noun phrase should refer to an entity using a definite description (*the Brazilian singer*), a pronoun (*she*) or a proper name (*Elis Regina*).



(a) *genre*



(b) *influence*

Figure 1.1: Syntactic trees for (1.1a) *genre* with verb phrase in the past tense (*PST*), simple aspect (*S*), indicative mood (*IND*) and active voice (*ACT*); and (1.1b) *influence* with verb phrase in the past tense (*PST*), simple aspect (*S*), indicative mood (*IND*) and passive voice (*PASS*).

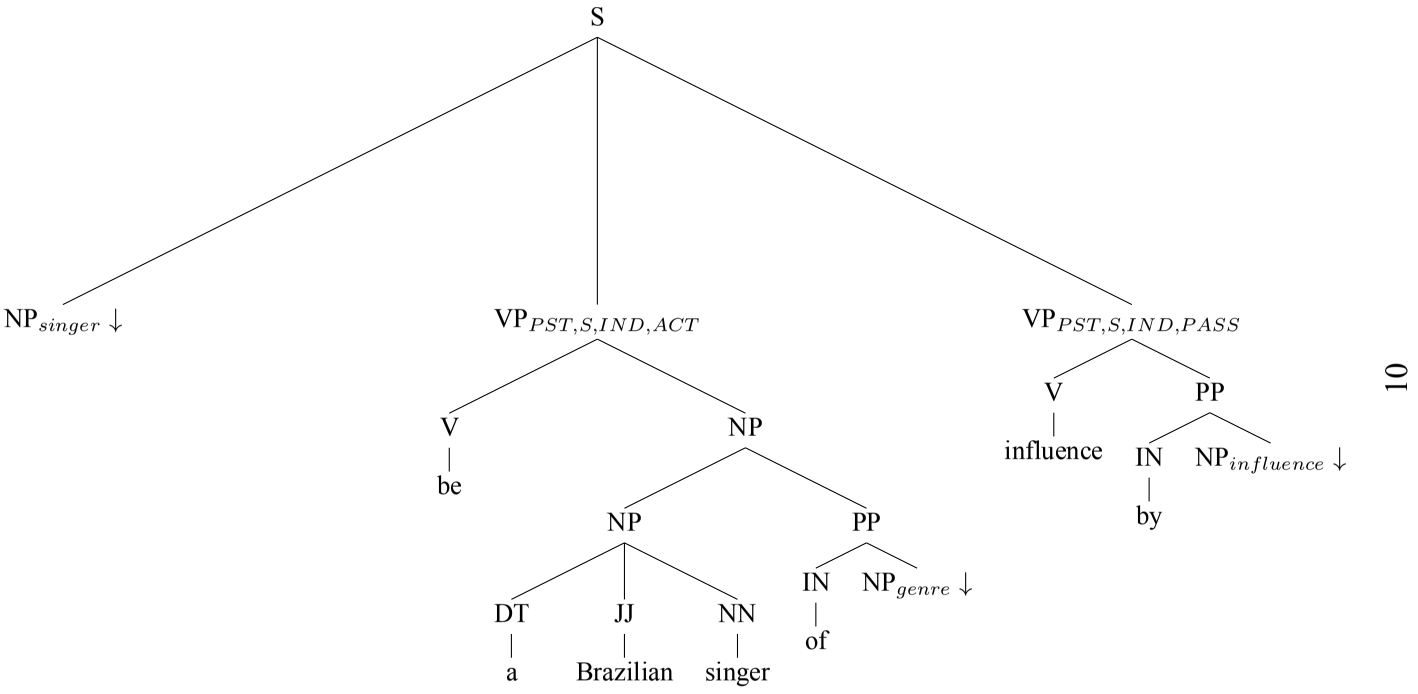
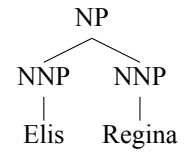
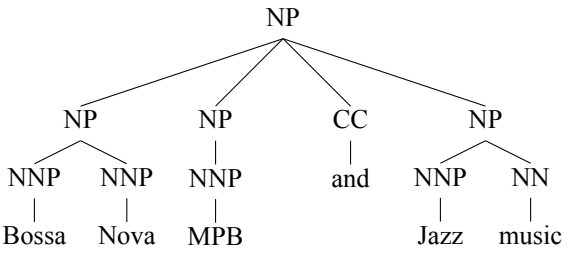


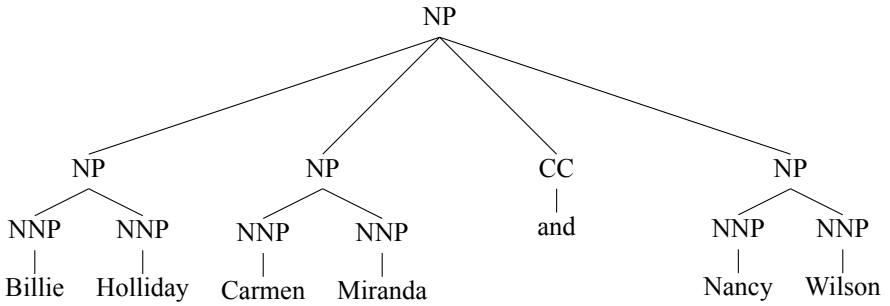
Figure 1.2: Aggregating syntactic trees for *genre* and *influence* triples.



(a) NP$_{singer}$ ↓



(b) NP$_{genre}$ ↓



(c) NP$_{influence}$ ↓

Figure 1.3: Referring expressions for (1.3a) NP$_{singer}$ ↓, (1.3b) NP$_{genre}$ ↓ and (1.3c) NP$_{influence}$ ↓.

Once the referential form has been decided, the content needs to be selected as well, choosing among different possible descriptions (*the singer*, *the Brazilian woman*, etc.) (e.g., Dale & Reiter, 1995; Krahmer & Theune, 2002; Krahmer et al., 2003), pronouns (*she*, *her*, *hers*, etc.) (e.g., Henschel et al., 2000; Callaway & Lester, 2002) or proper names (*Elis*, *Regina*, *Elis Regina*, etc.) (e.g., Siddharthan et al., 2011; van Deemter, 2016).

Let us say, for the sake of this example, that the system has decided to simply generate a full proper name referring to *Elis_Regina* (as depicted in figure 1.3), and that similarly full NPs are generated for her genre and influences. In this way the gaps can be filled, and the final stage, Textual Realization, can proceed.

Textual Realization

After selecting the content to be generated, structuring this into paragraphs and planning the individual sentences, Textual Realization aims to perform the last steps of converting the non-linguistic data into text. This includes setting the verbs in their right format according to tense, aspect, mood and voice (if the system knows Elis Regina died in 1982 for instance, past tense may be used), as well as the agreement between nouns and verbs. Often, realizers also need to insert function words (such as auxiliary verbs and prepositions) and punctuation marks (Gatt & Krahmer, 2018).

Once this has been taken care of, the system can produce, for example, the following short summary for our example:

> "Elis Regina was a Brazilian singer of Bossa Nova, MPB and Jazz Music, influenced by Billie Holiday, Carmen Miranda and Nancy Wilson."

The text may have been realized differently depending on the output

of the previous tasks of the pipeline. For instance, in case both clauses were not aggregated in Sentence Planning as Figure 1.2 depicts, the text would be realized as:

> "Elis Regina was a Brazilian singer of Bossa Nova, MPB and Jazz Music. Elis Regina was influenced by Billie Holiday, Carmen Miranda and Nancy Wilson."

This is grammatical, but does not 'flow' so nicely because of the repeated name. Alternatively, the text can also become more coherent when for the second reference to *Elis_Regina* a pronoun is chosen by the REG model rather than a full proper name:

> "Elis Regina was a Brazilian singer of Bossa Nova, MPB and Jazz Music. She was influenced by Billie Holiday, Carmen Miranda and Nancy Wilson."

1.1.2 Data-driven NLG

In recent years, various more data-driven approaches to NLG have been proposed, tackling many of the tasks in the pipeline architecture. For example, for the Document Planning phase, Barzilay & Lapata (2005) introduced a method to learn content selection rules in the football domain from a parallel corpus of documents and a corresponding database, in which the entries that should appear in each document were marked. From unannotated documents within a domain, Barzilay & Lee (2004) adapted a Hidden Markov Model to structure a text in terms of the topics a text addressed and the order in which these topics appear.

In Sentence Planning, Reiter et al. (2005) trained decision trees to learn how to lexicalize numerical weather data into English time phrases. Barzilay & Lapata (2006) described an approach based on Integer Linear Programming (ILP) to learn aggregation rules from a text and its related database whereas Bayyarapu (2011) models the aggregation task as a hypergraph partitioning problem. In the Referring Expression Generation literature, several data-driven models have been proposed to solve different parts of the process, including the choice of referential form (Belz et al., 2010), content selection for description generation (Viethen & Dale, 2010; Castro Ferreira & Paraboni, 2014) and proper name generation (Siddharthan et al., 2011).

Although models engineered in a modular structure such as Reiter & Dale (2000) can perform well in particular domains, it turns out to be difficult to adapt them to other ones (Angeli et al., 2010). Recently, data-driven models have been proposed which perform Content Selection and Surface Realization in a more integrated approach.

In Surface Realization, Belz (2008) introduced a probabilistic grammar to generate forecast text from weather data. The model was trained and evaluated on the SUMTIME corpus (Reiter et al., 2005) and managed to generate forecasts that were rated higher than human-produced ones in a human evaluation. Lu et al. (2009) trained and evaluated Tree Conditional Random Fields on the GEOQUERY (Kate et al., 2005) and ROBOCUP (Chen & Mooney, 2008) corpora in order to textually realize geographical queries and soccer statistics, respectively. Wen et al. (2015) proposed a neural generative model with semantically-conditioned Long Short-Term Memory layers (LSTM) to textually realize information about hotel and restaurant venues, and Wen et al. (2016) used a model in the same style to also realize information about laptops and televisions. Lebret et al. (2016)

and Chisholm et al. (2017) also proposed neural models to generate biographical sentences from fact tables from Wikipedia biographies, whereas Dong et al. (2017) introduced a neural model to textually realize product reviews.

Other studies proposed end-to-end NLG models to perform Content Selection and Surface Realization in a single, integrated framework instead of splitting both tasks. For instance, given a set of sentences and a set of logical meaning representations (MRs) describing soccer events based on the ROBOCUP corpus (Chen & Mooney, 2008), Kim & Mooney (2010) introduced an approach to align each meaning representation to its respective sentence for the purpose of later developing a semantic parser for content selection and a generative probabilistic model for surface realization, based on the alignments. Angeli et al. (2010) presented a generative model for content selection and surface realization trained and tested on ROBOCUP, SUMTIME (Reiter et al., 2005) and WEATHERGOV (Liang et al., 2009) corpora. Konstas & Lapata (2013) introduced a global model for automatic text generation based on a probabilistic context-free grammar trained and evaluated on three domains: sportcasting with ROBOCUP corpus, weather forecasts with WEATHERGOV corpus and the flight domain with the ATIS corpus (Dahl et al., 1994). Finally, Mei et al. (2016) proposes a neural end-to-end NLG model trained and evaluated on the ROBOCUP and WEATHERGOV corpora.

1.1.3 Interim summary

NLG systems aim to convert non-linguistic data into a linguistic output. They often work by first deciding *what* to say followed by choosing *how* to say it, steps commonly known as Content Selection and Surface Realization. We have briefly described the core components of these two tasks,

using a toy example involving summaries of Brazilian female singers. For more details on the individual tasks, the reader can consult, for example, Reiter & Dale (2000).

In recent years, the field of NLG research has gradually evolved from a focus on rule-based modular systems towards more data-driven approaches, not seldomly in an integrated end-to-end strategy (Gatt & Krahmer, 2018). In this thesis, we focus on developing data-driven models for two distinct NLG problems. The first problem concerns developing data-driven models for Referring Expression Generation (REG) which take linguistic variation into account in order to automatically generate more varied texts. The second problem concerns comparing different data-driven strategies for NLG based on two popular and distinct meaning representations. We introduce both topics in more detail below.

1.2 Generating varied text in NLG

Different from traditional NLG systems, humans have no problem in producing varied texts to describe a given communicative goal. In order to improve its *humanlikeness*, our first aim in this thesis is taking linguistic variation into account in the NLG process.

Previous NLG approaches that aimed to generate varied outputs often focused on taking pragmatic variation into account, generating different texts as a function of the (pragmatic) context. As pointed out by Bateman (1997), "NLG proper is not so much concerned with the generation of *a* text, but more with the generation of *the* text that is most appropriate for its context". An example of such a system includes the one developed by Bateman & Paris (1989) which tackled deciding what to say and how say it for specific audiences. In a similar vein, Hovy (1990) introduced

PAULINE, an NLG system able to reproduce text tailored to the hearer and the situation by handling variables about the conversational atmosphere, the speaker, the hearer and their relationship. More recent systems include Dong et al. (2017), who generated product reviews personalized for each user of the system, and Andreas & Klein (2016) who generated rich, contextually appropriate descriptions of structured world representations using neural networks that process the context visually and textually.

Besides studies that explore linguistic variation from a pragmatic point of view, there are also studies looking into "individual variation", aiming to quantify the relative frequency of lexical and syntactic choices made by different humans *in the same context*. As an example, Smiley et al. (2016) collected a corpus with verb choices made by different authors to describe the rising and falling behavior of a stock, like the frequency with which the verbs *rocketed up* and *jumped up* were used to describe a rising pattern (e.g., *GoPro's stock* ***rocketed up*** *19 percent* and *GoPro's stock* ***jumped up*** *19 percent*), revealing that there is indeed variation among writers' choices in a same situation.

In this current thesis, we explore how to generate more varied text with an emphasis on Referring Expression Generation (REG). Initially, we look into traditional modular REG approaches, in which a model is first used to choose the form of a reference followed by another model which decides on the content of the referring expression given the chosen form. In this particular approach, we aim to generate varied references exploring two subtasks of the process: the choice of referential form and proper name generation. Finally, we introduce a novel end-to-end REG approach which produces varied referring expressions in discourse, simultaneously deciding on form and content.

Choice of referential form Despite the large number of algorithms which have been proposed for the choice of referential form (Reiter & Dale, 2000; Henschel et al., 2000; Callaway & Lester, 2002; Krahmer & Theune, 2002; Gupta & Bandopadhyay, 2009; Greenbacker & McCoy, 2009), all of these are deterministic, always choosing the same referential form in the same discourse context, which does not allow for much variation. This happens partly because these models are often based on corpora that have only one gold standard per situation. To illustrate the problem of training and evaluating models based on one gold standard corpus, consider the highlighted referring expressions in the following text as gold standard references to the topic:

Elis Regina$_1$ was **a Brazilian singer**$_1$ of Bossa Nova, MPB and Jazz Music. **She**$_1$ did not play any instrument.

Let us now say that a model for referential form selection chooses a *proper name* as the form of the third reference, which might subsequently be realized as "Elis". In comparison with the gold standard in the corpus - a *pronoun* - the choice of the model would count as an error. However, the use of a pronoun does not necessarily mean that the choice of a proper name would be wrong in that case, since it arguably does not affect the quality of the rest of the text and other writers might have chosen this form as well. Given that texts in a corpus contain only one gold standard in each situation (the choice of the writer), it is hard to estimate how much individual variation there could have been in that particular context.

In order to solve these problems, we introduce a new corpus, which we dubbed VaREG, where we collect gold standards of 20 different writers for each reference to the topic of a text. Based on this new dataset, we introduce data-driven models, trained and evaluated on the corpus, that

model potential linguistic variation in the choice of referential form, and we study what impact this has on the variation and appreciation of generated texts.

Proper name generation In modular REG approaches, besides variation in the choice, we also address variation in the realization of one particular referential form: proper names. Proper name generation has received almost no attention in the literature, even though it is frequently used to refer to named entities in text. Often it is assumed that merely using the full proper name will suffice (Reiter & Dale, 2000), but van Deemter (2016) has shown that this strategy has a number of problems. As a notable exception, Siddharthan et al. (2011) suggested two manual rules for proper name generation: include a *full name* in a initial reference in discourse, and "use surname only, remove all pre- and post-modifiers" for subsequent references to a same entity. However, the problem is not as straightforward as this solution seems to suggest. Names in the *full name* form may vary depending on the named entity to be referred to. For some people, the combination of first and last birth names counts as a *full name* (like *Marisa de Azevedo Monte*), whereas for others, the combination of *first* and *middle* birth names would apply (like *Elis Regina Carvalho Costa*). Moreover, using the surname for discourse-old references may not always work well either (like for *Elis Regina* and *Marisa Monte*).

In fact, we know very little about how proper names are produced in text, and how much variation there actually is. To find out, we first collected a new corpus, which we call REGnames: a dataset with 53,102 proper name references to 1,000 people in different discourse contexts. Based on REGnames, we develop a statistical model able to generate variations of a proper name by taking into account pragmatic information like

the person to be mentioned and the discourse context as well as individual variation.

End-to-End REG is different from traditional REG approaches like the ones previously addressed. In the traditional architecture, approaches to REG rely on features extracted from the discourse and focus either on selecting referential form (Orita et al., 2015), or on selecting referential content, typically zooming in on one specific kind of reference such as a pronoun (e.g., Henschel et al., 2000; Callaway & Lester, 2002), definite description (e.g., Dale & Haddock, 1991; Dale & Reiter, 1995) or proper name generation (e.g., Siddharthan et al., 2011; van Deemter, 2016). Going in a different direction, as the last study of this part of the thesis, we introduce a novel end-to-end REG approach, relying on deep neural networks, which makes decisions about form and content in one go without explicit feature extraction. This model works by constructing representations of the surrounding linguistic context, which are later used in the generation of a group of varied referring expression candidates that are likely to suit the given context.

1.3 Generating natural language from what?

Besides varied outputs, the second core theme of this thesis concerns the *input* to NLG systems. As said above, while there is broad consensus among scholars on the output of NLG systems (i.e., text or speech), there is far less agreement on what the input should be. Over the years, NLG systems have taken a wide range of inputs, including for example images, numeric data and semantic representations.

Among data-driven models, semantic representations are the most popular kind of input. However, even for this particular kind, there is no

```
temperature(time=5pm-6am,min=48,mean=53,max=61)
windSpeed(time=5pm-6am,min=3,mean=6,max=11,mode=0-10)
windDir(time=5pm-6am,mode=SSW)
gust(time=5pm-6am,min=0,mean=0,max=0)
skyCover(time=5pm-9pm,mode=0-25)
skyCover(time=2am-6am,mode=75-100)
precipPotential(time=5pm-6am,min=2,mean=14,max=20)
rainChance(time=5pm-6am,mode=someChance)
```

Figure 1.4: Semantic Representation as a set of *event-attribute-values* for the sentence *"A 20 percent chance of showers after midnight. Increasing clouds, with a low around 48 southwest wind between 5 and 10 mph"*. `WEATHERGOV` instance extracted from Angeli et al. (2010).

agreement upon a single format. Data-driven models have been developed to generate text from a wide range of semantic representations which vary from each other in terms of level of specification and domain restrictions (Angeli et al., 2010; Kim & Mooney, 2010; Konstas & Lapata, 2013; Wen et al., 2015; Lebret et al., 2016; Mei et al., 2016; Wen et al., 2016; Chisholm et al., 2017; Dong et al., 2017).

Some data-driven models have generated texts from very detailed se-

Attribute	Value
TITLE	mathias tuomi
SEX OR GENDER	male
DATE OF BIRTH	1985-09-03
OCCUPATION	squash player
CITIZENSHIP	finland

Figure 1.5: Semantic Representation as a set of attributes for the sentence *"Mathias Tuomi, (born September 30, 1985 in Espoo) is a professional squash player who represents Finland."*. Instance extracted from Chisholm et al. (2017).

mantic input representations, relatively *close* to the intended linguistic output format and, sometimes, even containing some syntactic and lexical information. Koller & Striegnitz (2002), for instance, modeled NLG as a dependency parsing problem. White & Rajkumar (2008) and Gyawali & Gardent (2014) covered the problem using Combinatory Categorial Grammar (CCG) and Tree Adjoining Grammars (TAG), respectively. Finally, Belz et al. (2011) proposed generating text from shallow and deep representations extracted from the Penn Tree Bank (PTB) (Marcus et al., 1994). In general, models which require very specific input representation have a somewhat limited general applicability, and are relatively difficult to adapt from one domain to the next.

Other data-driven NLG systems work with input representations that are less complex, and not so close to the linguistic output, simply focusing on semantics. For example, some models generate text from sets of *event-attribute-values* (Angeli et al., 2010; Konstas & Lapata, 2013; Wen et al., 2015; Mei et al., 2016; Wen et al., 2016) (Figure 1.4), attributes (Lebret et al., 2016; Chisholm et al., 2017; Dong et al., 2017) (Figure 1.5), triples (Kim & Mooney, 2010) (e.g., the representations in Figures 1.1 and 1.2) and other structures like trees (Lu et al., 2009). Although models that convert this kind of representations into text are more general, the NLG process in this scenario involves making more choices, which may increase the complexity of the decision process.

The fact that different NLG systems rely on very different input formats is clearly a limitation: it is more difficult to compare performances of different approaches, and acts as a barrier for exchanging insights and technical implementations. As a result, researchers have started looking for candidate input formats that could be used more broadly within the community. These inputs should be sufficiently detailed to allow for in-

teresting and high quality output generation, but at the same time not too detailed to hinder general applicability. Recently two candidates have started to become popular: Abstract Meaning Representation (Banarescu et al., 2013) and RDF Triples from the Semantic Web (Bizer et al., 2009). In this thesis we will study generation from both these semantic input representations.

AMRs are structures that encode the meaning of a sentence as a rooted, directed and acyclic graph, where nodes represent concepts, and labeled directed edges represent relations among these concepts (Banarescu et al., 2013). Besides semantics, these representations also distinguish entities by their Wikipedia IDs (wikified references) and represent syntactic and lexical information. In terms of resources, when this thesis was written (end 2017, early 2018), LDC2017T10 contained the largest AMR corpus with 39,260 AMR-sentence pairs in the domains of newswire, discussion forums, web logs and television transcripts. Other examples are "Bio AMR" and "Little Prince", corpora in the biomedical and novel domains, respectively. The former has 6,452 pairs, whereas the latter consists of 1,562 pairs. For more information about AMR corpora, the website of the meaning representation can be consulted*.

Resource Description Framework (RDF) is perhaps the best known protocol in the Semantic Web (a machine-friendly extension of the World Wide Web). Each RDF unit is in the triple format, consisting of a Subject, Predicate and Object (e.g., `Alan Bean | occupation | Test pilot`). Based on this representation, the WebNLG was a challenge organized for automatically converting a set of RDF triples into English text (Gardent et al., 2017a,b). For the challenge, a parallel corpus was provided with 25,298 instance pairs in 15 domains: Astronaut, University, Monument,

*`https://amr.isi.edu/download.html`

Building, Comics Character, Food, Airport, Sports Team, Written Work, City, Athlete, Artist, Mean of Transportation, Celestial Body and Politician.

In this thesis we study the generation from both AMR and RDF-triples. In order to convert the meaning representations into text, we develop a data-driven model based on modular NLG systems (Reiter & Dale, 2000) and compare it against new models we developed, making use of Machine Translation approaches (both phrase-based and neural ones) (Koehn et al., 2003; Bahdanau et al., 2015). A comparison between these different models is a secondary goal of this part of the thesis.

1.4 This thesis

This thesis consists of two parts, one looking at varied outputs (Chapters 2-6) and one zooming in on semantic inputs (Chapters 7-8).

In the first part, we aim to take linguistic variation into account in the NLG process, focusing on the Referring Expression Generation task. By collecting and analyzing new corpora of referring expressions, we are initially able to develop new data-driven models for two subtasks in modular REG: the choice of referential form and proper name generation. Later, we introduce an end-to-end approach, based on neural networks, which generates varied referring expressions to a discourse entity, deciding on its form and content in one shot.

In the second part of this work, we focus on generating text from two recent meaning representation formats. In both representations, the content had been already selected and the focus is on surface realization. We propose NLG models based on a pipeline architecture as well as models that work in a less modular style, by using methods from Statistical (Koehn

et al., 2003) and Neural (Bahdanau et al., 2015) Machine Translation.

This thesis is based on articles; each chapter is self-contained, with its own introduction and discussion. As a result, a small amount of overlap between chapters was unavoidable. Similarly, due to differing formats and different requests from reviewers and editors, some small changes in presentation style between chapters may occur.

Chapter 2 focuses on data for individual variation in referential choice, collecting and analyzing a new corpus (VaREG). For this data collection, we presented different writers with texts in which all references to the main topic of the text have been replaced with gaps. The task of the participants was to fill each of those gaps with a reference to the topic. In total 9,588 referring expressions are collected in this way, produced by 78 different participants for 563 referential gaps - around 20 referring expressions per reference - in 36 English texts (equally divided over three genres: comparing encyclopedic texts, news articles and product reports). In the analysis, we estimated to what extent different writers agree with each other in terms of normalized entropy. In addition, we study whether this variation depends on the text genre. The annotated corpus is made publicly available.

Chapter 3 introduces two different models that take individual variation into account for the choice of referential form: a Naive Bayes and a Recurrent Neural Network. Based on an automatic evaluation using the VaREG corpus, we choose the best performing model to be used in the process of generating referring expressions to discourse topics of texts from the GREC-2.0 corpus. In a human evaluation, the coherence and comprehensibility of the texts with the generated references by the model were compared with the original texts as well as with a version of the texts whose the references were generated by a random baseline model.

Chapter 4 describes the collection and analysis of REGnames, a corpus for the study of proper name references in discourse. It consists of 53,102 proper names references to 1,000 people in 15,241 webpages. In the analysis of the corpus, we aim to identify the different ways, in terms of length and form, in which proper names are produced along discourse.

Chapter 5 presents two versions of a new probabilistic model of proper name generation: one that always chooses the most likely proper name form and one that relies on a "roulette wheel" selection model to generate more varied references. These models rely both on the nature of the entity referred to (what is the likelihood that a given person will be referred to using, say, the first or last name?) and on the discourse context for generating proper name references in text. In an intrinsic evaluation experiment, we compare the performance of the two versions of this model with our implementations of three baselines. We also describe a human evaluation experiment where we compare original texts with alternative versions that include proper names generated by our model.

Chapter 6 introduces NeuralREG: an end-to-end approach addressing the full Referring Expression Generation task, which given a number of entities in a text, produces corresponding referring expressions, simultaneously selecting both form and content. The approach is based on neural networks which generate referring expressions to discourse entities relying on the surrounding linguistic context, without the use of any feature extraction technique. In an automatic and human evaluation, we compared our novel approach against two baselines, relying on a specific constructed set of 78,901 referring expressions to 1,501 entities in the context of the semantic web, derived from a (delexicalized) version of the WebNLG corpus. Both the data set and the model are publicly available.

Chapter 7 focusses on the process of automatically generating text

from Abstract Meaning Representation (AMR), framing it as a translation task and comparing two different MT approaches (Phrase-based and Neural MT). We also look at potential benefits of three preprocessing steps on AMRs before feeding them into an MT system: *delexicalization*, *compression*, and *linearization*. Delexicalization decreases the sparsity of an AMR by removing constant values, compression removes nodes and edges which are less likely to be aligned to any word on the textual side and linearization 'flattens' the AMR in a specific order. Combining all possibilities gives rise to $2^3 = 8$ AMR preprocessing strategies. Following earlier work in AMR-to-text generation and the MT literature, we automatically evaluate the system's outputs in terms of fluency, adequacy and post-editing effort.

Chapter 8 presents a comparison between modular and end-to-end approaches to automatically generate text from RDF triples, a popular protocol from the Semantic Web. Our modular approach performs the task in 4 sequential steps: discourse ordering, template selection, referring expression generation and text reranking. For the end-to-end approaches, we introduce Statistical and Neural Machine Translation models to convert the non-linguistic data from the Semantic Web into English text. We evaluated our models based on the results in the WebNLG challenge, where an automatic and human evaluation were conducted. In the automatic evaluation, metrics were computed to measure fluency, adequacy and post-editing effort of the output texts. In the human evaluation, human judges evaluated their quality according to semantics, grammaticality and fluency.

Finally, Chapter 9 summarizes the findings of this thesis and discusses the main topics addressed like modeling variation, semantic representations, the comparison of modular against end-to-end approaches, and eval-

uation in NLG. Finally, we point to some topics which could possibly be addressed in future work.

2

Variation in the choice of referential form: A corpus study

Abstract In this chapter, we aim to measure the variation between writers in their choices of referential form by collecting and analysing a new and publicly available corpus of referring expressions*. The corpus consists of referring expressions produced by different participants in identical situations. Results, measured in terms of normalized entropy, reveal substantial individual variation. We discuss the problems and prospects of this finding for automatic text generation applications.

*https://ilk.uvt.nl/~tcastrof/vareg/

This chapter is based on Castro Ferreira, T., Krahmer, E., & Wubben, S. (2016). Individual variation in the choice of referential form. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, NAACL-HLT'2016 (pp. 423-427). San Diego, California: Association for Computational Linguistics.

2.1 Introduction

Automatic text generation is the process of automatically converting data into coherent text - practical applications range from weather reports (Goldberg et al., 1994) to neonatal intensive care reports (Portet et al., 2009). One important way to achieve coherence in texts is by generating appropriate referring expressions throughout them (Krahmer & van Deemter, 2012). In this generation process, the choice of referential form is a crucial task (Reiter & Dale, 2000): when referring to a person or object in a text, should the system use a proper name ("Phillip Anschutz"), a definite description ("the American entrepreneur") or a pronoun ("he")?

Despite the large amount of algorithms developed for deciding upon the form of a referring expression (Callaway & Lester, 2002; Greenbacker & McCoy, 2009; Gupta & Bandopadhyay, 2009; Orăsan & Dornescu, 2009; Greenbacker et al., 2010), it is difficult to know how well these algorithms actually perform. Typically, such algorithms are evaluated against a corpus of human written texts, predicting what form each reference should have in a given context. Now consider a situation in which the algorithm predicts that a reference should be a description, while this same reference is a pronoun in the corpus text. Should this count as an error? The answer is: it depends. The use of a pronoun does not necessarily mean that the use of a description is incorrect. In fact, other writers might have used a description as well.

In general, corpora of referring expressions have only *one* gold standard referential form for each situation, while different writers may conceivably vary in the referential form they would use. This complicates the development and evaluation of text generation algorithms, since these will typically attempt to predict the corpus gold standard, which may not always be representative of the choices of different writers. Although re-

cent work in text generation has explored individual variation in the content determination of definite descriptions (Viethen & Dale, 2010; Castro Ferreira & Paraboni, 2014), to the best of our knowledge this has not been systematically explored for choosing referential forms.

In this paper, we collect and analyze a new corpus to address this issue. In the collection, we presented different writers with texts in which all references to the main topic of the text have been replaced with gaps. The task of the participants was to fill each of those gaps with a reference to the topic. In the analysis, we estimated to what extent different writers agree with each other in terms of normalized entropy. In addition, we study whether this variation depends on the text genre, comparing encyclopedic texts with news and product reports. Moreover, we discuss the implications of our findings for automatic text generation, exploring whether factors such as syntactic structure, referential status and recency affect the variation between the writers' choices. The annotated corpus is made publicly available[†].

2.2 Data gathering

2.2.1 Material

For our study, we used 36 English texts, equally distributed over three different genres: news texts, reviews of commercial products and encyclopedic texts. The encyclopedic texts were selected from the GREC corpus (Belz et al., 2010), which is a standard corpus for testing and evaluating models for choice of referential form. The news and review texts were selected from the AQUAINT-2 corpus[‡] and the SFU Review corpus

[†] http://ilk.uvt.nl/~tcastrof/vareg

[‡] http://catalog.ldc.upenn.edu/LDC2008T25

(Konstantinova et al., 2012), respectively.

Note that, depending on the genre, texts may address different kinds of topics. For instance, the news texts usually are about a person, a company or a group; the product reviews may be about a book, a movie or a phone; and the encyclopedic texts about a mountain, a river or a country. In all texts, all expressions referring to the topic were replaced with gaps, which the participants should fill in.

2.2.2 Participants

Participants were recruited through CrowdFlower[§]. 78 participants completed the survey. 53 were female and 25 were male. Their average age was 37 years old. Most were native speakers (73 participants) or fluent in English (5 participants).

2.2.3 Procedure

The participants were first presented with an introduction to the experiment, explaining the procedure and asking their consent. Next, they were asked for their age, demographic information and English language proficiency. After this, participants were randomly assigned to a list, containing 9 texts (3 per genre).

The task of the participants was to fill in each gap with a reference to the topic of the text. To inform the participants about the entities, a short description - extracted from the Wikipedia page about the topic - was provided before each text.

Participants were encouraged to fill in the gaps according to their preferences, so that they felt the texts would be easy to understand. We made

[§] http://www.crowdflower.com/

sure that participants did not fill all the gaps in a text with only one referring expression (to avoid copy/paste behavior). Participants could also not leave any gap empty (they were instructed to use the "-" symbol for empty references).

2.2.4 Annotation

The author of this thesis annotated the referring expressions produced by participants for referential form, syntactic position, referential status, and recency. Coding was straightforward, and the few difficult cases were resolved in discussions between the co-authors.

The referring expressions were assigned to one of five forms: **proper names** ("*Philip Anschutz*, 66, will have no trouble keeping busy."); **pronouns** ("*It* is the highest peak [...]", "Huffman, *who* spoke at the sentencing phase [...]"); **definite descriptions** ("[...] *the Russian President* defended the country's contribution [...]"); **demonstratives** ("You'll probably have screaming kids who want to see *this movie*."); and **empty references** ("He rarely grants on-the-record media interviews and __ seldom allows himself to be photographed.").

Following the GREC Project scheme (Belz et al., 2010), referring expressions were annotated for three syntactic positions: subject noun phrases, object noun phrases, and genitive noun phrases that function as determiners (***Google's*** *stock*). Referential status refers to whether a referring expression is a first mention to the topic (new) or not (old). We annotated this at the level of the text, paragraph and sentence, so that a reference can be new in paragraph, but old in the text. Recency, finally, is the distance between a given referring expression and the last, previous reference to the same topic, measured in terms of number of words within a paragraph. If the referring expression was the first mention to the topic

in the paragraph, its recency is set to 0.

In total, 10,977 referring expressions were collected in 563 referential gaps. 3,682 were annotated as proper names, 4,662 as pronouns, 768 as definite descriptions, 318 as demonstratives and 158 as empty references. The remaining 1,389 were ruled out of the corpus, since they did not consist of a reference to the target entity or changed the meaning of the original sentence.

2.2.5 Analysis

We measured variation between participants' choices for each gap, using the normalized entropy measure, defined in Equation 2.1, where X corresponds to the references in a given gap, and $n = 5$ the number of referential forms annotated.

$$H(X) = -\sum_{i=1}^{n=5} \frac{p(x_i)\log(p(x_i))}{\log(n)} \tag{2.1}$$

The measure ranges from 0 to 1, where 0 indicates the complete agreement among the participants for a particular referential form, and 1 indicates the complete variation among their choices.

2.3 Results

Figure 2.1 presents the main result, depicting the amount of individual variation in referential forms, measured in terms of entropy, as a function of text genre. The averaged entropies are significantly higher than 0 for all three genres according to a Wilcoxon signed-rank test (News: $V = 20,910.0$, $p < .001$; Reviews: $V = 11,476.0$, $p < .001$; and Encyclopedic texts: $V = 10,153.0$, $p < .001$). This clearly shows that

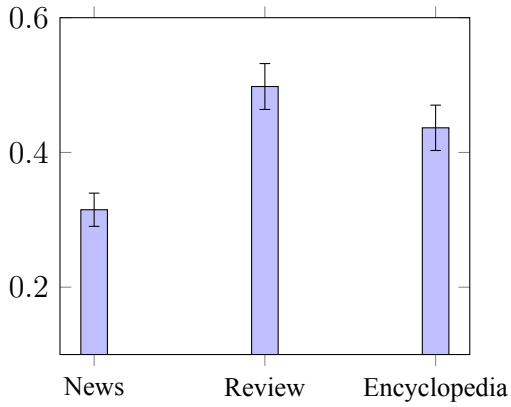


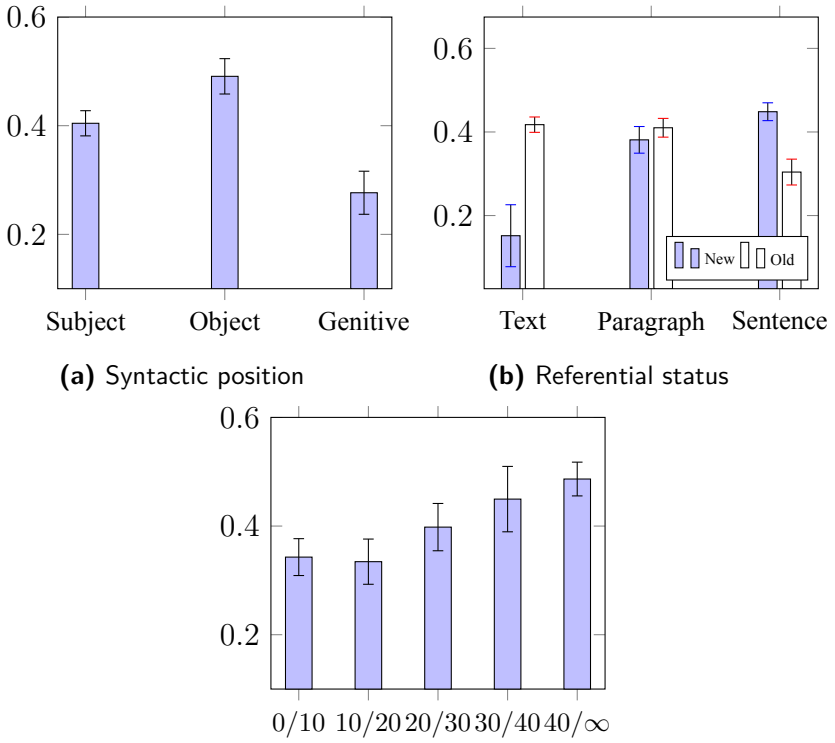
Figure 2.1: Average entropy per gap as a function of text genre. The error bars represent the 95% confidence intervals.

different writers can vary substantially in their choices for a referential form. Comparing the three different genres, we find that writers' choices of referential form varied most in review texts and least in news texts, with encyclopedic texts sandwiched in between (Kruskal-Wallis $H = 70.73$, $p < .001$).

In comparison with the original texts, 44% of the referring expressions produced by the writers differ from the original ones in a same referential gap. Furthermore, the form of the original referring expressions differs from the major choice of the writers in 38% of the referential gaps.

To get a better understanding of factors potentially influencing individual variation, we investigate the effects of three linguistic factors: syntactic position, referential status and recency. Figure 2.2 depicts the average entropies for each of these.

Comparing the three syntactic positions, Figure 2.2a suggests that the highest variation is found when writers need to choose referential forms in the object position of a sentence, whereas the lowest variation is found



(a) Syntactic position **(b)** Referential status

(c) Recency

Figure 2.2: Average entropy per gap as a function of: (2.2a) syntactic position, (2.2b) referential status, (2.2c) recency. Error bars represent 95% confidence intervals. In Figure 2.2c, the bars represent the average entropies for the group of references where the most recent prior reference is 10 or less words away, between 11 and 20 words, between 21 and 30 words, between 31 and 40 words and more than 40 words away.

for references that function as a genitive noun phrase determiner (Kruskal-Wallis $H = 52.53, p < .001$).

Figure 2.2b depicts individual variation in the choice of referential form for old and new references in the text, paragraph and sentence. The data suggests a higher amount of individual variation when writers need to refer to a topic already mentioned in the text rather than a first mention (Mann-Whitney $U = 3,916$, $p < .001$), presumably because for a topic which is new in the text, writers were more likely to agree to use proper names (91% of the choices). Looking at old and new references within paragraphs reveals no significant differences in individual variation (Mann-Whitney $U = 32,669.5$, $p < .094$). At the sentence level, finally, there is more individual variation for references to a new topic than for references to a previously mentioned one (Mann-Whitney $U = 21,873.0$, $p < .001$). When writers referred to a previously mentioned referent in the sentence, they tended to agree on the use of a pronoun (76% of the choices).

Figure 2.2c shows the individual variation in referential form as a function of recency. Except for the relatively nearby intervals (between 0 and 10 words, and between 11 and 20 words), the data suggests that when the distance between two consecutive references gets larger, the variation among writers' choices increases (Kruskal-Wallis $H = 35.31, p < .001$).

2.4 Discussion

In this paper, we studied individual variation in the choice of referential form by collecting a new (and publicly available) dataset in which different participants (writers) were asked to refer to the same referent throughout a text. This was done for different genres (news, product review and

encyclopedic texts) by measuring the variation between participants in terms of normalized entropy. If participants would all use the same referential form in the same gap, we would expect entropy values of 0 (no individual variation), but instead we found a clearly different pattern in all three text genres. Moreover, we also saw a considerably difference in form among the original referring expressions and the ones generated by the participants. This reveals that substantial individual variation between writers exists in terms of referential form.

To get a better understanding of which factors influence individual variation, we analyzed to what extent three linguistic factors had an impact on the entropy scores: syntactic position, referential status and recency. We found a higher amount of individual variation when writers had to choose referential forms in the direct object position, referring to previously mentioned topics in the text and first mentioned ones in the sentence, and references that were relatively distant from the most recent antecedent reference to the same topic.

These findings can be related to theories of reference involving the salience of a referent (Gundel et al., 1993; Grosz et al., 1995, among others). Brennan (1995), for example, argued that references in the role of the subject of a sentence are more likely to be salient than references in the role of the object. Chafe (1994), to give a second example, pointed out that references to previously mentioned referents in the discourse and ones that are close to their antecedent are more likely to be salient than references to new referents or ones that are distant from their antecedents. Note, incidentally, that none of these earlier studies address the issue of individual variation in referential form.

Arguably, the amount of individual variation is even larger than the data reported here suggest. To illustrate this, consider, for instance, that

different participants referred to *Phillip Frederick Anschutz* - the main topic of one of the texts used - as *Phillip Frederick Anschutz*, *Mr. Phillip Frederick Anschutz*, *Anschutz*, *Mr. Anschutz* and *Phillip Anschutz*. Even though these all have the same referential form (proper names), there is also a lot of variation *within* this category. Indeed, it would be interesting in future research to explore which factors account for this within-form variation.

The current findings are important for automatic text generation algorithms in two ways. First, they are beneficial for developers of text generation systems, since they allow for a better understanding of the range of variation that is possible in referring expression generation. Second, they allow for a more principled evaluation of algorithms predicting referential form. In fact, the collected corpus paves the way for developing models which predict frequency distributions over referential forms, rather than merely predicting a single form in particular context (as current models do).

3

Variation in the choice of referential form: Data, models and evaluation

Abstract In this chapter, we describe two non-deterministic models, a Naive Bayes model and a Recurrent Neural Network, that account for individual variation in the choice of referential form in automatically generated texts. Both models are evaluated using the VaREG corpus. Then we select the best performing model to generate referential forms in texts from the GREC-2.0 corpus and conduct an evaluation experiment where humans judge the coherence and comprehensibility of the generated texts, comparing them both with the original references and those produced by a random baseline model. Data and models are publicly available*.

*https://github.com/ThiagoCF05/ReferentialForm

This chapter is based on 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*, ACL'2016 (pp. 568–577). Berlin, Germany: Association for Computational Linguistics.

3.1 Introduction

Automatic text generation is the process of converting non-linguistic data into coherent and comprehensible text (Reiter & Dale, 2000). In recent years, interest in text generation has substantially increased, due to the emergence of new applications such as "robot-journalism" (Clerwall, 2014). Even though computers these days are perfectly capable of automatically producing text, the results are arguably often rather rigid, always producing the same kind and style of text, which makes them somewhat "boring" to read, especially when reading multiple texts in succession.

Human-written texts, by contrast, do not suffer from this problem, presumably because human authors have an innate tendency to produce variation in their use of words and constructions. Indeed, psycholinguistic research has shown that when speakers produce referring expressions in comparable contexts, they non-deterministically vary both the form and the contents of their references (Dale & Viethen, 2010; Van Deemter et al., 2012). In this paper, we present and evaluate models of referring expression generation that mimic this human non-determinacy and show that this enables us to generate varied references in texts, which, in terms of coherence and comprehensibility, did not yield significant differences from human-produced references according to human judges.

In particular, in this chapter we focus on the choice of referential *form*, which is the first decision to be made by referring expression generation models (Reiter & Dale, 2000) and which determines whether a reference takes the form of a proper name, a pronoun, a definite description, etc. Several such models have been proposed (Reiter & Dale, 2000; Henschel et al., 2000; Callaway & Lester, 2002; Krahmer & Theune, 2002; Gupta & Bandopadhyay, 2009; Greenbacker & McCoy, 2009). However, all of

these are fully deterministic, always choosing the same referential form in the same context.

The fact that these models are generally based on text corpora which have only one gold standard form per reference (the one produced by the original author) does not help either. When the corpus contains, say, a description at some point in the text, this does not mean that, for example, a proper name could not occur in that position as well (Yeh & Mellish, 1997; Castro Ferreira et al., 2016a). Generally, we just don't know. To counter this problem, a recent corpus, called VaREG, was developed in which 20 different writers were asked to produce references for a particular topic in a variety of texts, giving rise to a distribution over forms per reference (Chapter 2; Castro Ferreira et al., 2016a). This gives us the possibility to distinguish situations where there is more or less agreement between writers in their choices for a referential form. But it also enables a new paradigm for choosing these forms, where instead of predicting the most likely one, we can in fact predict the frequency in which a reference assumes a specific form, allowing us to turn the choice of referential form into a non-deterministic probabilistic model.

In this chapter, we introduce two different models that take the individual variation into account for the choice of referential form, one based on Naive Bayes and one on Recurrent Neural Networks. Both are evaluated using the VaREG corpus. Furthermore, we use the best performing model to generate referential forms in texts from the GREC-2.0 corpus, based on the roulette-wheel generation process (Belz, 2008), and conduct an evaluation experiment in which humans judge the coherence and comprehensibility of the generated texts, comparing them both with the original references and those produced by a random baseline model.

3.2 Related work

Several models for the choice of referential form have been proposed in the literature. They can roughly be distinguished in two groups: rule-based and data-driven models.

Many rule-based models were created for pronominalization, i.e, to choose whether an object or person should be referred to using a pronoun or not. Reiter & Dale (2000) proposed one of the first rule-based models, which opts for a pronominal reference only if the referent was previously mentioned in the discourse and no mention to an entity of same gender can be found between the reference and its antecedent. Henschel et al. (2000) presented a pronominalization model based on recency, discourse status, syntactic position, parallelism and ambiguity. To decide among a pronoun or a definite description, Callaway & Lester (2002) also proposed a rule-based model which makes the choices based on information about the discourse, rhetorical structure, recency and distance. Krahmer & Theune (2002) extended the Incremental algorithm so that if a referent achieves a level of salience in the discourse (measured by a salience weight), a pronoun is used. Otherwise, a definite description is produced to distinguish the referent from the distractors.

Aiming to make choices similar to humans, some studies proposed machine learning models trained on human choices of referential form. The GREC project (Belz et al., 2010) motivated the development of many of those data-driven models. One of the project's shared tasks aimed to predict the form of the references to the main topics of texts taken from Wikipedia. Among the participants of the task, Gupta & Bandopadhyay (2009) presented a model that combined rules and a machine learning technique based on semantic and syntactic category, paragraph and sentence positions, and reference number. Similarly, Greenbacker & McCoy

(2009) proposed a decision tree that, besides the features used in Gupta & Bandopadhyay (2009), was also based on recency and part-of-speech features. For more information on the GREC shared task, see Belz et al. (2010).

One limitation that these models all have in common is that they fail to model individual variation. According to their predictions, a reference will always assume the most likely referential form. For example, a model that takes into account syntactic position will always choose the same referential form for the subject of a sentence, while humans tend to vary in their choices of referential form. One of the reasons for this problem arises from the data these models are trained on. Most corpora only contain one referring expression per reference. Only the newly introduced VaREG corpus takes variation into account, containing 20 different expressions for each reference, allowing us to model distributions over referential slots.

3.3 The VaREG corpus

The VaREG corpus was collected for the study of individual variation in the choice of referential form (Chapter 2; Castro Ferreira et al., 2016a). The corpus is based on a number of texts, which were presented to participants in such a way that all references to the main topic of the text had been replaced with gaps. Each participant was asked to fill each of those gaps with a referring expression to the topic.

The resulting corpus consists of 9,588 referring expressions, produced by 78 participants for 563 referential gaps - around 20 referring expressions per reference - in 36 English texts. The texts were equally distributed over 3 genres: news texts, reviews of commercial products and encyclopedic texts. The references were annotated according to their syntactic

position (subject, object, etc.), referential status (new or old, in text, paragraph and sentence) and recency (number of words between previous reference to the same object or entity). Moreover, the referring expressions of the participants were classified into 5 referential forms: proper names, pronouns, definite descriptions, demonstratives and empty references.

The analysis of the corpus revealed considerable variation among participants in their choices of referential forms. Various factors influenced the amount of variation that occurred. High amounts of variation, for example, were found in product reviews and also in the object position of sentences. Besides allowing us to distinguish between situations with relatively high and relatively low individual variation in choices of referential form, this corpus introduces a new paradigm for the development and evaluation of models for referential choice. Rather than predicting the most likely form of a reference, as is usually done, the new corpus allows us to develop a model that can predict the frequency with which a particular reference can assume different referential forms. In this chapter, we explore this possibility.

3.4 Models

We model the individual variation in the choice of referential form in the following way: each reference consists of a tuple (X, y), where X is the set of feature values that describes the reference and y is a distribution of referential forms that indicates the frequency (in proportion) in which X assumes each form. So given X, we expect to find a distribution $\hat{y}$ similar to y.

Table 3.1 depicts the features used to describe X. The influence of those discourse factors in the choice of referential form has been often

Feature	Description
Syntactic position	Subject, object or a genitive noun phrase in the sentence.
Referential Status	First mention to the referent (new) or not (old) in text, paragraph and sentence.
Recency	Distance between a given reference and the last, previous one to the same referent.

Table 3.1: Features used to describe the references.

studied in the literature. Concerning syntactic position, Brennan (1995) argued that references in the subject position of a sentence are more likely to be shorter than references in the the object position. In favor of status and recency, Chafe (1994) showed that references to previously mentioned referents in the discourse and ones that are close to their antecedents are more likely to be shorter than references to new referents or ones that are distant from their antecedents.

All features were defined categorically, including recency. This latter was treated by describing if a reference's antecedent is 10 or less words away, between 11 and 20 words, between 21 and 30 words, between 31 and 40 words and more than 40 words away.

To predict a distribution $\hat{y}$ based on X, we propose two models: a Naive Bayes and a Recurrent Neural Network.

3.4.1 Naive bayes

Given a set of referential forms F, the probability that a reference assumes a particular form $f \in F$ according to this model is given by:

$$P(f \mid X) \propto \frac{P(f) \prod_{x \in X} P(x \mid f)}{\sum_{f' \in F} P(f') \prod_{x \in X} P(x \mid f')} \tag{3.1}$$

To avoid zero probabilities, we used additive smoothing with $\alpha = 2e^{-308}$. So given a reference described by X, $\hat{y}$ is the distribution over F:

$$\hat{y} = \begin{bmatrix} P(f_1 \mid X) \\ ... \\ P(f_{|F|} \mid X) \end{bmatrix} \tag{3.2}$$

3.4.2 Recurrent neural network

Some referential theories support the idea that a referential form is chosen based on previous choices to the same referent. Arnold (1998) argued that subjects of a sentence are more likely to be later pronominalized, as well as references in parallel syntactic position with their antecedents. Chafe (1994) sustained that referents mentioned in recent clauses also tend to be pronominalized. Since Naive Bayes does not take into account the sequential nature of text, we use a Recurrent Neural Network (RNN) to be able to take context into account. RNN is a powerful structure to handle sequences of data, which can map a sequence of references $(X_1, ..., X_t)$ to their referential forms distributions $(y_1, ..., y_t)$ based on the previous steps.

Our approach here is similar to the one presented by Mesnil et al. (2013). But instead of word continuous representations, a referential embedding is created for each combination of feature values in X. So given a reference X_t and a context window size win, the embeddings of the references $X_{t-win/2}^{t-1}$, X_t and $X_{t+1}^{t+win/2}$ are merged to form a representation e_t. This representation is used in equations 3.3 and 3.4 to find a distribution over the referential forms that X_t could assume.

$$h_t = sigmoid(W^{he} e_t + W^{hh} h_{t-1}) \tag{3.3}$$

$$\hat{y_t} = softmax(W^{yh}h_t) \tag{3.4}$$

We assume a sequence of tuples $\{(X_1, y_1)..., (X_t, y_t)\}$ as all the references to a referent throughout a text. The RNN was trained using Backpropagation Through Time. To measure the error among y and $\hat{y}$, we used cross entropy as a cost function. The values for the remaining parameters of the RNN are introduced in Table 3.2. We chose them based on an ad-hoc analysis, where we searched for an optimal combination to obtain the best predictions.

Batch Size	10
Context Window Size	3
Epochs	15
Embedding Dimension	50
Hidden Layer Size	50
Learning Rate	0.1

Table 3.2: RNN Settings

3.5 Individual variation experiments

For each reference slot encountered in the VaREG corpus, we evaluated how well a model takes the individual variation into account in the choice of referential form by comparing its predicted distribution of referential forms ($\hat{y}$) with the real distribution (y). We performed this comparison through two experiments.

In the first, the models were trained and tested with VaREG corpus. In the second, we aimed to check to what extent the referring expressions from the GREC-2.0 corpus are similar in form to the referring expressions

from VaREG corpus by training the models with the first corpus and testing with the second.

3.5.1 Method

4-fold-cross-validation was used to train the models in the first experiment. The number of folds was chosen based on the set-up of the VaREG corpus, which consists of 4 groups of texts. Given the structure of the corpus, we decided that training our model with 3 groups of texts and testing it on the held-out group was the most natural solution to avoid overfitting. Each fold has the same amount of texts per genre.

Unlike VaREG, GREC-2.0 corpus does not have a set of referring expressions for the exact same reference. So, in the second experiment, the referential form distributions y were defined globally by grouping the references by X and computing the frequency of each referential form.

We also re-annotated the GREC-2.0 corpus to make it compatible with the VaREG corpus. In particular, we added features for status and recency to the former and made the terminology consistent between the two corpora[†]. Both the VaREG corpus and the re-annotated GREC-2.0 corpus are publicly available[‡].

3.5.2 Metrics

For each reference, Jensen-Shannon divergence (Lin, 1991) was used to measure the similarity between y and $\hat{y}$:

$$JSD(y||\hat{y}) = \frac{1}{2}D(y||m) + \frac{1}{2}D(\hat{y}||m) \tag{3.5}$$

[†]Texts also used in VaREG had their references removed from the GREC-2.0 version used here.

[‡]http://ilk.uvt.nl/~tcastrof/acl2016

where $m = \frac{1}{2}(y + \hat{y})$

In this measure, D is the Kullback-Leibler divergence (Kullback, 1997). The Jensen-Shannon divergence ranges from 0 to 1, in which 0 indicates full convergence of the two distributions and 1 full divergence. Therefore, a lower number indicates a better individual variation modeling.

To check the behavior of $\hat{y}$ based on y in each reference, the referential forms of both distributions were ranked and their relation were analyzed with the Spearman's rank correlation coefficient. This measure ranges between -1 and 1, where -1 indicates a fully opposed behavior among the variables and 1 the exact same behavior among them. 0 indicates a non-linear correlation among the involved variables.

3.5.3 Baselines

We considered two baseline models in the experiments. The first, called *Random*, assumes $\hat{y}$ as a random distribution of forms for each reference.

The second model, called *ParagraphStatus*, always chooses a proper name when the reference is to a new topic in the paragraph (the distribution will assume the value 1 to the proper name form and 0 to the others), and a pronoun otherwise (value 1 to the pronoun form and 0 to the others).

3.5.4 Results

Cross-validation on VaREG corpus

Table 3.3 depicts the Jensen-Shannon divergence and Spearman's correlation coefficient of the models cross-validated on VaREG corpus. All our models outperformed the baselines.

Considering the models in which the references are described by only one kind of feature, it seems that the status features (+Status) are the ones

Models	JSD	$\rho_{y,\hat{y}}$
Random	0.63	-0.01
ParagraphStatus	0.43	0.66
NB+Syntax−Status−Recency	0.39	0.69
NB−Syntax+Status−Recency	0.32	**0.75**
NB−Syntax−Status+Recency	0.41	0.68
NB+Syntax+Status−Recency	**0.31**	**0.75**
NB+Syntax−Status+Recency	0.38	0.70
NB−Syntax+Status+Recency	0.33	0.73
NB+Syntax+Status+Recency	**0.31**	0.74
RNN+Syntax−Status−Recency	0.37	0.71
RNN−Syntax+Status−Recency	0.36	0.72
RNN−Syntax−Status+Recency	0.40	0.70
RNN+Syntax+Status−Recency	0.33	0.73
RNN+Syntax−Status+Recency	0.37	0.71
RNN−Syntax+Status+Recency	0.36	0.72
RNN+Syntax+Status+Recency	0.33	0.72

Table 3.3: Average Jensen-Shannon divergence and Spearman's correlation coefficient of the models in Experiment 1.

that best contributed to model the individual variation in the choice of referential form, whereas the recency (+Recency) is the worst. Syntactic position is sandwiched among the previous two.

In the comparison within Naive Bayes and RNN models, the ones in which the references are described by syntactic position and referential status (+Syntax+Status−Recency) obtained the best results for both measures. Figure 3.1 depicts the average Jensen-Shannon divergences by genre of Naive Bayes and RNN models in which the references are described by this combination of features. Both models presented the best results in encyclopedic texts, and the worst in product reviews.

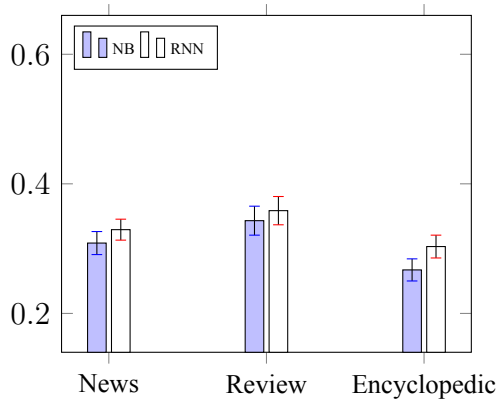


Figure 3.1: Jensen-Shannon divergence of NB+Syntax+Status−Recency (NB) and RNN+Syntax+Status−Recency (RNN) by genre in Experiment 1. Error bars represent 95% confidence intervals.

Although RNNs are able to model the individual variation in a reference based on its antecedents, they did not introduce significantly better results than Naive Bayes. In fact, NB+Syntax+Status−Recency is significantly better than RNN+Syntax+Status−Recency in modeling the individual variation in news (Wilcoxon $Z = 11,574.5$, $p < 0.01$) and encyclopedic texts (Wilcoxon $Z = 4,232.5$, $p < 0.001$).

Training on GREC-2.0 and evaluating on VaREG corpus

Table 3.4 shows the results of models trained with GREC-2.0 and tested with VaREG corpus. These models are the two versions of Naive Bayes, and the two versions of RNN which were best evaluated in the previous experiment.

The results of this experiment follow the results of the previous one. Our models outperformed the baselines and NB+Syntax+Status−Recency was the model that obtained the best results for both measures.

Models	JSD	$\rho_{y,\hat{y}}$
Random	0.63	-0.01
ParagraphStatus	0.43	0.66
NB+Syntax+Status−Recency	**0.36**	**0.67**
NB+Syntax+Status+Recency	0.37	0.64
RNN+Syntax+Status−Recency	0.37	0.62
RNN+Syntax+Status+Recency	0.37	0.64

Table 3.4: Average Jensen-Shannon divergence and Spearman's correlation coefficient of the models in Experiment 2.

Figure 3.2 depicts the Jensen-Shannon divergence measures of models NB+Syntax+Status-Recency and RNN+Syntax+Status-Recency by text genre. As in the previous experiment, both Naive Bayes and RNN models best modeled the individual variation in encyclopedic texts. Moreover, there was not significant difference among NB+Syntax+Status-Recency and RNN+Syntax+Status-Recency in the three text genres.

In general, the models trained with VaREG corpus seemed to model the individual variation in the choice of referential form better than the models trained with GREC-2.0 corpus.

3.6 Coherence and comprehensibility of the texts

In this section, we investigate to what extent texts generated by our non-deterministic method are judged coherent and comprehensible by readers. We do this by comparing texts from the GREC-2.0 corpus in which all references were (re)generated using our method, with the original text and with a variant that includes random variation of referential form.

Version	Text
Original	**Spain**, officially the Kingdom of Spain, is a country located in Southern Europe, with two small exclaves in North Africa (both bordering Morocco). **Spain** is a democracy **which** is organized as a parliamentary monarchy. **It** is a developed country with the ninth-largest economy in the world. **It** is the largest of the three sovereign nations that make up the Iberian Peninsula–the others are Portugal and the microstate of Andorra.
Random	**It**, officially the Kingdom of Spain, is a country located in Southern Europe, with two small exclaves in North Africa (both bordering Morocco). **The country** is a democracy **that** is organized as a parliamentary monarchy. **It** is a developed country with the ninth-largest economy in the world. **This country** is the largest of the three sovereign nations that make up the Iberian Peninsula–the others are Portugal and the microstate of Andorra.
Generated	**Spain**, officially the Kingdom of Spain, is a country located in Southern Europe, with two small exclaves in North Africa (both bordering Morocco). **Spain** is a democracy **that** is organized as a parliamentary monarchy. **The country** is a developed country with the ninth-largest economy in the world. **It** is the largest of the three sovereign nations that make up the Iberian Peninsula–the others are Portugal and the microstate of Andorra.

Table 3.5: Example of text in the Original, Random and Generated version.

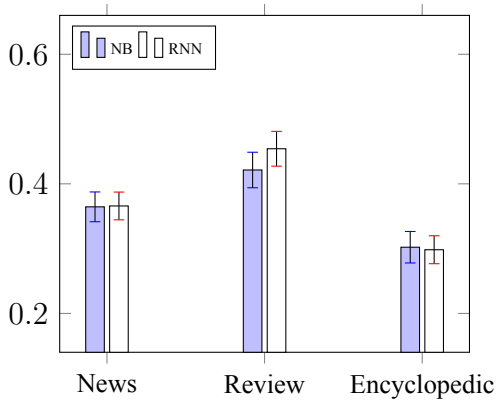


Figure 3.2: Average Jensen-Shannon divergence of NB+Syntax+Status−Recency (NB) and RNN+Syntax+Status-Recency (RNN) by genre in Experiment 2. Error bars represent 95% confidence intervals.

3.6.1 Our model for choice of referential form

To generate the referring expressions for the topic of a given text of GREC-2.0, we first grouped all references by syntactic position and referential status values. Then for each group, we shuffled the references and chose their forms according to the distribution predicted by our best performing model (the NB+Syntax+Status−Recency trained on VaREG). The choice of referential forms followed the roulette-wheel generation process (Belz, 2008). This process entails that if a group has 5 references and our model predicts a distribution of 0.8 proper names and 0.2 pronouns, 4 references of the group will be proper names and 1 a pronoun.

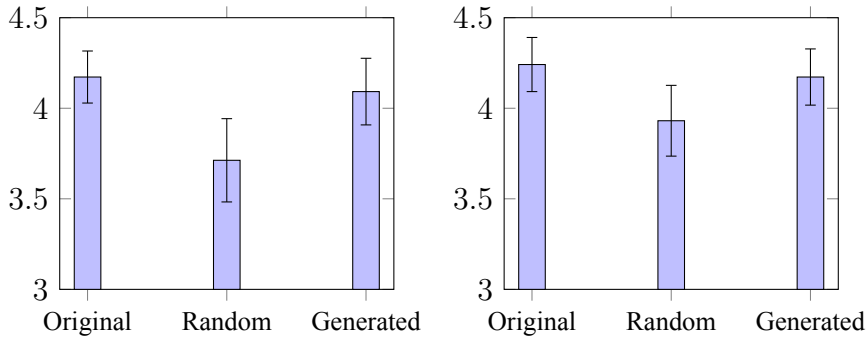
This covers the selection of referential forms (deciding which form to use at which particular point in the text). To deal with their linguistic realization, we implemented the following heuristics. For the cases in which a proper name reference was selected, we chose a realization depending on referential status. If the reference was the first mention to the topic in the

text, the reference was realized with the topic's longest proper name. Otherwise, the reference was realized with its shortest proper name. For the cases in which a definite description was selected, but where the original GREC-2.0 corpus did not provide a description for the topic, we selected the shortest predicate adjective of the first sentence of the text, immediately following the main verb. For instance, for the sentence *"Alan Mathison Turing was an English mathematician, logician, and cryptographer."*, the selected definite description would be "**The English mathematician**". In the cases where a reference should assume the form of a demonstrative, the definite article of the definite description was replaced by the demonstrative "this" (In the previous example, "**This English mathematician**").

3.6.2 Evaluation method

We evaluated three versions of each text. The *Original* is the original text in the corpus, including the original referring expressions selected by the author. We compared this version with a *Random* variant, which does include variation of referential forms, but selecting them in a fully random way. Finally, in the third, *Generated* version, all references were generated according to the method outlined at Section 3.6.1. Table 3.5 depicts an example of text in the three versions.

In total, we make 3 versions of 9 pseudo-randomly selected texts (5 covering animate topics and 4 inanimate ones, varying in length) from the GREC-2.0 corpus, yielding 27 texts in total. These were distributed over 3 lists, such that each list contained one variant of each text, and there was an equal number of texts from the 3 conditions (*Original*, *Random*, *Generated*). In all texts, all references to the topic were highlighted in yellow. The experiment was conducted on CrowdFlower and is publicly



(a) Coherence **(b)** Comprehensibility

Figure 3.3: Average coherence (3.3a) and comprehensibility (3.3b) of the texts with the original, randomized and generated referring expressions. Error bars represent 95% confidence intervals.

available[§].

The experiment was performed by 30 participants (10 per list). Their average age was 36 years, and 22 were female. All were proficient in English (the language of the experiment), 26 participants were native speakers. They were asked to rate each text in terms of how coherent and comprehensible they considered it, on a scale from 1 (Very Bad) to 5 (Very Good).

3.6.3 Results

Figure 3.3 depicts the average coherence and comprehensibility of the texts where their topics are described by the *Original*, *Random* and *Generated* approaches, respectively. Inspection of this Figure clearly shows that the *Random* texts are rated lower than both the *Original* and the *Generated* texts, and that the latter are rated very similarly on both dimensions.

[§] http://ilk.uvt.nl/~tcastrof/acl2016

This is confirmed by the statistical analysis. According to a Friedman test, there is statistically significant difference in the coherence ($\chi^2 = 11.79$, $p < 0.005$) and comprehensibility ($\chi^2 = 8.98$, $p = 0.01$) for the three kinds of texts. We then conducted a post hoc analysis with Wilcoxon signed-rank test corrected for multiple comparisons using the Bonferroni method, resulting in a significance level set at $p < 0.017$. Texts of the *Original* approach are statistically more coherent ($Z = 322$, $p < 0.017$) and comprehensible ($Z = 407.5$, $p < 0.017$) than texts of the *Random* one. Texts of the *Generated* approach are also statistically more coherent ($Z = 275$, $p < 0.017$), but not more comprehensible ($Z = 378$, $p < 0.05$) than texts of the *Random* one. Finally, and crucially, comparing *Original* and *Generated* texts revealed no significant differences for coherence ($Z = 540$, $p < 0.5$) nor for comprehensibility ($Z = 391.5$, $p < 0.5$).

3.7 Discussion

In this paper we explored the possibilities of introducing more variation in automatically generated texts, by trying to model individual variation in the selection of referential form. We relied on a new corpus (VaREG, Chapter 2; Castro Ferreira et al., 2016a), which does not contain a single expression for each reference in a text, but rather a distribution of referential forms produced by 20 different people. In contrast to earlier models for referential choice which always deterministically choose the most likely form of a reference, we proposed a Naive Bayes and a Recurrent Neural Network model which aimed to predict the frequency distribution with which a reference can assume a specific referential form, based on discourse features including syntactic position, referential status and recency. Given a reference, we evaluated how well each different model could cap-

ture the individual variation found in the VaREG corpus by comparing its predicted distribution of referential forms with the real one in the corpus. We trained the models in two different ways: first using the VaREG, and second using the GREC-2.0 corpus. The Naive Bayes model, trained on VaREG corpus, in which the references were described by syntactic position and referential status features was the one that best modeled the individual variation in the choice of referential form.

Features Referential status features were the most helpful for modeling the individual variation in the choice of referential form. They were followed by the syntactic position feature. Both of these findings are consistent with the observations about human variation in the selection of referential forms, as discussed by Castro Ferreira et al. (2016a) (Chapter 2). This chapter argued that writers are more likely to vary in their choices when a reference is in the object position, and when it is an old mention in the text, but new in the sentence. Recency was not a helpful feature for our models, and this may be due to the way the feature was represented - i.e., as a categorical rather than a continuous feature. Moreover, the recency feature was measured in terms of words between the current reference and the most recent previous one to the same referent. Perhaps, it would be better to measure recency in terms of different discourse entities mentioned between two references to the same referent.

Genre In agreement with Castro Ferreira et al. (2016a) (Chapter 2), we also found that genre mattered. For modeling variation, our models performed best when applied to encyclopedic texts, and worst in product reviews, with news sandwiched in between.

Naive Bayes model vs. RNNs Although the RNNs were able to model individual variation in the choice of referential form to some extent, they did not perform significantly better than the Naive Bayes models, which might have to do with the relatively small dataset. However, we think the size of the corpus matches the relatively low complexity of the problem we address. In the most complex case (i.e., when a reference is described by its syntactic position, status and recency), an input can be represented in 120 different ways to predict a multinomial distribution of size 5 (number of referential forms). This complexity is much smaller than other problems typically modeled by RNNs. In text production, for instance, an input may be represented by thousands of words to predict a large multinomial distribution over a vocabulary (Sutskever et al., 2014). Additionally, it is important to stress that we actually have a real multinomial distribution to compare with the distribution predicted by the RNN in each situation. We observed that it is possible to compute more fine-grained error costs in our case, which makes the RNN converge faster when it is backpropagated. In sum, we believe that those two factors combined compensate for the size of the dataset. A possible explanation for the non-difference among the Naive Bayes model and RNNs is the use of the referential status features, which perhaps are already enough to model the relation among a reference and its antecedents.

VaREG corpus vs. GREC-2.0 corpus Interestingly, our proposed models yielded better performance when trained on the VaREG than on the GREC-2.0 corpus. This shows a difference among the referential choices of both corpora. We conjecture this difference is partly due to differences in text genres, since the VaREG corpus contains texts from three different genres, whereas the GREC-2.0 corpus only has encyclopedic texts. Ear-

lier work has also highlighted the influence of text genre on the amount of individual variation in writers' choices for referential forms (Chapter 2; Castro Ferreira et al., 2016a).

Coherence and comprehensibility In the second part of the study, we used the best performing model to generate referential forms in texts from the GREC-2.0 corpus, using a roulette-based model sampling from the predicted distributions over referential forms. We evaluated the texts generated in this way in an experiment in which humans were asked to judge the coherence and comprehensibility of the generated texts, comparing them both with the original references and those produced by a random baseline model. In terms of coherence and comprehensibility, we found that the texts in which the references were generated by our model were not significantly different than the human generated ones, and significantly better than the randomly generated ones. This shows that our solution does not only model the individual variation in the choice of referential form, but that this also does not negatively affect the quality of the texts. This is an important step towards developing new models for automatic text generation that are less predictable and more varied.

4

Variation in proper name generation: A corpus study

Abstract In this chapter, we introduce a corpus for the study of proper name generation. The corpus consists of proper name references to people in webpages, extracted from the Wikilinks corpus. In our analyses, we aim to identify the different ways, in terms of length and form, in which proper names are produced throughout a text. The corpus is publicly available*.

*http://ilk.uvt.nl/~tcastrof/regnames/

This chapter is based on Castro Ferreira, T., Wubben, S., & Krahmer, E. (2016). Towards proper name generation: a corpus analysis. In *Proceedings of the 9th International Natural Language Generation conference*, INLG'2016 (pp. 222-226). Edinburgh, Scotland: Association for Computational Linguistics.

4.1 Introduction

In natural language generation systems, referring expression generation (REG) is the process of producing references to discourse entities (Krahmer & van Deemter, 2012). Among the referential forms which can be used to distinguish an entity, proper names are an important and commonly used one. For instance, Castro Ferreira et al. (2016a) (Chapter 2) showed that writers produce a proper name as a first mention to an entity in 91% of the cases.

In generation systems, not only the choice of whether a proper name should be generated is important, but also which *form* the proper name should take. For instance, *Barack Hussein Obama II* is the birth name of the 44th president of United States of America. However, he is also commonly referred to as *Barack Obama*, *Obama*, *President Obama*, etc. How to automatically decide which form to use?

In this paper, we introduce a new corpus of 53,102 proper names referring to people in 15,241 texts[†]. We analyse the corpus in terms of distribution of proper name lengths, intuitively expecting an inversely proportional relation between length of a name and sentence number in a text. We also analyze these references in terms of the presence of the first, middle and last name of the entity; and whether the reference is accompanied by a title or an appositive.

4.2 Related work

Unlike the generation of descriptions (Krahmer & van Deemter, 2012), only a few studies have focussed on the automatic generation of proper

[†] https://ilk.uvt.nl/~tcastrof/regnames

names. Reiter & Dale (2000), for instance, suggests the use of a full proper name for initial reference, optionally followed by an appositive to indicate properties of the entity important for the discourse. However, their approach does not account for variation in proper name references.

van Deemter (2014) argues that proper name variants can be generated using standard algorithms for the generation of descriptions. In other words, the study proposes to represent proper names as a set of attribute-value pairs extracted from a knowledge base. Just like a description set with the attribute-value pairs $\{(type, cube), (color, blue)\}$ may be generated to single out a target from different colored objects, a proper name set like $\{(firstName, Frida), (lastName, Kahlo)\}$ can be generated to single out a person from others with different names in a context set. Van Deemter, however, does not apply this model in the context of text generation.

Siddharthan et al. (2011) presented a model to (re)generate referring expressions to people in extractive summaries. When generating a proper name, the model chooses between a full name (*Frida Kahlo*) or only a surname (*Kahlo*). Moreover, it also decides whether to use pre- (role, affiliation and temporal modifiers) or post-modifiers (appositives and relative clauses). As far as we know, this is the only study that introduced a corpus analysis of how humans produce proper names in a discourse. However, it only distinguished proper names among full names and surnames in a small set of 876 news texts.

4.3 Data gathering

4.3.1 Materials

To analyze how proper names are used in text, we analyzed webpages from the Wikilinks corpus (Singh et al., 2012). This corpus was originally created to study cross-document coreferences and comprises around 40 million mentions to 3 million entities. All the mentions were extracted automatically by finding hyperlinks to Wikipedia pages related to the entities.

To collect our data, we identified the 1,000 most frequently mentioned people in the corpus. To determine which entities are persons, we used DBpedia, a database that provides structured information from Wikipedia (Bizer et al., 2009). From the Wikilinks corpus, we then randomly chose a subset of webpages that contain at least one mention to one of the most frequently mentioned persons. In total, our corpus contains texts from 15,241 webpages.

4.3.2 Annotation

To annotate the proper name references, we created a knowledge base which describes all variations of a proper name for the studied persons. We also parsed the webpages to identify in which part of the discourse the different proper name references were used. The annotation procedure is explained in more detail below.

Proper Names Knowledge Base We used two ontologies present on DBpedia to extract different proper names for the studied entities. The FOAF (*Friend-of-a-Friend*) ontology was used to extract the name (foaf:name), the given name (foaf:givenName) and the surname

(foaf:surname) of a person. From the DBpedia ontology, we extracted the birth name of the entities (dbo:birthName).

Based on the proper names collected in DBpedia, we created a knowledge base by identifying 3 proper name attributes: **first name**, **middle name** and **last name**. First names consist of the first token from the name, given name and birth name, whereas last names consist of the token from the surname and the last tokens from the name and birth name. Middle names were defined as all the tokens which are neither the first token in the given and birth names nor the last token in the name and birth name. For instance, *Charles Bukowski* has *Charles*, *Bukowski*, *Charles Bukowski* and *Heinrich Karl Bukowski* as his given name, surname, name and birth name in DBpedia, respectively. Based on this information, the knowledge base for this entity would consist of *Charles* and *Heinrich* as first names; *Karl* as middle name; and *Bukowski* as last name.

Discourse Annotation The webpages were parsed using the Stanford CoreNLP software (Manning et al., 2014). Using this tool, we performed part-of-speech tagging, lemmatization, named entity recognition, dependency parsing, syntactic parsing, sentiment analysis and coreference resolution.

To improve the coreference resolution we performed a post hoc sanity check to see whether references which were labeled as being to the same entity were correct. For each entity distinguished by the software, we checked the proper nouns of each proper name reference. If at least the proper nouns of one proper name were values present in the knowledge base of the target entity, all the references of the entity distinguished by the software were considered references to the target entity.

Once the references to the target entity were distinguished, we anno-

tated their syntactic positions based on the output of the dependency parser and their referential statuses in the text and in the sentence - whether a reference is a first or an old mention to an entity. We also checked for the presence of a title or an appositive in the proper name references. These features were extracted based on the named entity recognition and dependency parser, respectively. In total, 53,102 proper name references were annotated in this way (an average of 3 per text).

4.3.3 Analyses

To analyze how proper names referring to people are distributed over a text, we checked the length of these references in terms of tokens. We also analyzed the possible variations of a proper name by checking the presence of the first, middle and last name of the entity, and whether the proper name was accompanied by a title or an appositive.

4.4 Results

Figure 4.1 depicts the average length of proper name references in the first 100 sentences of the texts. A linear regression clearly shows that the length of a proper name decreases along the text, as predicted.

Table 4.1 summarized the percentage of proper name attributes, revealing that the last name is the most used one, followed by first name. The others occur less frequently.

Figure 4.2 shows the average length of proper name references as a function of syntactic position and referential status. Proper names in the object role of a sentence are generally longer than those in subject position (a); proper names that are new in the text are longer than those that have

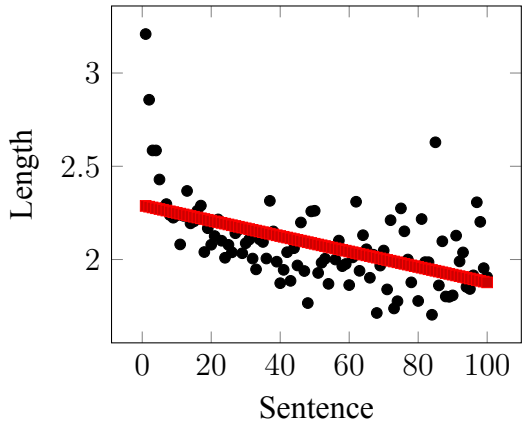
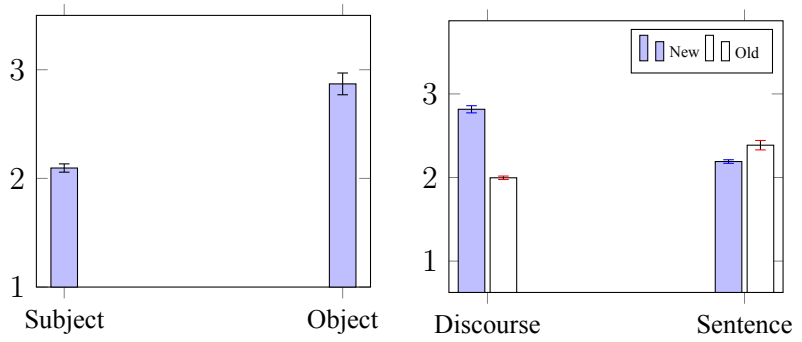


Figure 4.1: Average length of the proper names in tokens by sentence.

Title	2.4%
First Name	59.3%
Middle Name	7.1%
Last Name	89%
Appositive	1.7%

Table 4.1: Percentage of the proper name attributes



(a) Syntactic position **(b)** Referential status

Figure 4.2: Average length of the proper names as a function of: (4.2a) syntactic position and (4.2b) referential status. Error bars represent 95% confidence intervals.

been mentioned in the text before, and vice versa when looking at new/old references per sentence (b).

Table 4.2 depicts frequency of various attribute sets, as a function of syntactic position and referential status in the text and sentence. Proper names consisting of both first and last name are the most common in the corpus. This proper name form is also the most common one in the subject role of a sentence and as a mention to a new entity in the discourse. On the other hand, in the object role of a sentence and as mention to an old entity in the text, the use of only the last name is most common.

In general, proper names described by the first and last names (*First+Last*), and by the first, middle and last names (*First+Middle+Last*) occur more often in the subject role of a sentence as a mention to a new entity in the text. The combination of first and last names is also more likely as a mention to old entities in the sentence. Proper names described by just one proper name attribute reveal the opposite behavior, occurring more in the object role of a sentence as a mention to an old entity in the

	Syntax	
	Subject	Object
First+Last	57.41%	38.74%
Last	24.45%	37.17%
First	6.15%	11.98%
Middle+Last	3.39%	3.38%
First+Middle+Last	2.92%	2.79%
Middle	1.06%	1.88%
Others	4.62%	4.06%
	Text	
	New	Old
First+Last	69.52%	36.53%
Last	10.60%	44.26%
First	4.33%	10.12%
Middle+Last	4.62%	2.02%
First+Middle+Last	4.72%	1.36%
Middle	0.78%	1.74%
Others	5.43%	3.97%
	Sentence	
	New	Old
First+Last	44.19%	57.16%
Last	35.93%	26.61%
First	8.58%	7.78%
Middle+Last	2.91%	1.76%
First+Middle+Last	2.44%	1.53%
Middle	1.57%	0.80%
Others	4.38%	4.36%
	General	
First+Last	46.2%	
Last	34.9%	
First	8.5%	
Middle+Last	2.8%	
First+Middle+Last	2.3%	
Middle	1.5%	
Others	3.8%	

Table 4.2: Percentage of the attribute sets in the proper name references

text or new in the sentence.

4.5 Discussion

This chapter introduced a corpus for the study of proper name generation. We analyzed the different forms in which proper name references occur in text by checking their length as well as the occurrence of different proper name attributes including the first, middle, last names of the mentioned entity, as well as possible modifiers, such as titles or appositives.

Analyses revealed that longer proper names - in terms of number of tokens and proper name attributes - are more likely to be generated early in the text, in the object role of a sentence, and as the reference to a new entity in the text or an old in the sentence. Concerning referential status in text, our results are broadly in line with Siddharthan et al. (2011), which shows that a new entity in the text is more likely to be referred to the full name, whereas only the surname is used for an old entity. Concerning referential status in the sentence, the fact that a proper name reference to an old entity in the text is more likely to be longer than one to a new entity was somewhat unexpected, since some referential theories argue that a reference to previously mentioned entities tend to be shorter (Chafe, 1994). A possible explanation could be the presence of cataphoras, as in *Unlike **his** peers, **Harold Camping** does not pack a positive punch.*

As future work, we aim to develop a computational model for proper name generation based on the reported findings. Besides the variation between proper name forms in different parts of a text, this model should be able to address the proper name preferences for each entity. For instance, it should account that *Winston Churchill* is typically mentioned by his surname (*Churchill*), whereas *Napoleon Bonaparte* is by his first name

(*Napoleon*) in similar discourse contexts. We will address this by training individual models combining the a priori probability of a particular proper name for a particular individual with contextual factors. Additionally, we plan to annotate the proper name references to all the entities present in the texts of our corpus, and not only the references to the 1,000 people studied here. We think this expansion will give a broader view of the generation of proper names, since we will be able to study the process as a function of other discourse conditions, as topicality.

5

Variation in proper name generation: Data, models and evaluation

Abstract In this chapter, we introduce a statistical model able to generate variations of a proper name by taking into account the person to be mentioned, the discourse context and variation. The model relies on the REGnames corpus, a dataset with 53,102 proper name references to 1,000 people in different discourse contexts. We evaluate the versions of our model from the perspective of how human writers produce proper names, and also how human readers process them. The corpus* and the model† are publicly available.

*http://ilk.uvt.nl/~tcastrof/regnames/
†http://github.com/ThiagoCF05/ProperName

This chapter is based on Castro Ferreira, T., Krahmer, E., & Wubben, S. (2017). Generating flexible proper name references in text: Data, models and evaluation. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, EACL'2017 (pp. 655–664). Valencia, Spain: Association for Computational Linguistics.

5.1 Introduction

In automatic text generation, Referring Expression Generation (REG) is the task responsible for generating references to discourse entities, addressing, for example, the question whether the text should refer to an entity using a definite description (*the West Coast poet and patron saint of drinking writers*), a pronoun (*he*) or a proper name (*Henry Charles Bukowski*). REG is among the tasks which have received most attention in text generation (see Krahmer & van Deemter (2012), for a survey), but the vast majority of the research has concentrated on the generation of descriptions, while proper name generation has received virtually no attention, albeit with notable exceptions (Siddharthan et al., 2011; van Deemter, 2016) to which we return below.

Still, proper names occur frequently in texts. For instance, Castro Ferreira et al. (2016a) (Chapter 2) showed that human writers use proper names in 91% of the cases to initially refer to persons. Indeed, some earlier research on text generation has stated that discourse-new references should be generated by using the strategy to "simply give the name of the object (if it has a name)" (Reiter & Dale, 2000). However, the *Bukowski* example already indicates that this is not as straightforward as Reiter and Dale suggest - the poet's full name is *Henry Charles Bukowski* and his birth name is *Heinrich Karl Bukowski*, but he is more commonly known as simply *Charles Bukowski*; see also van Deemter (2016) for a discussion of this and other complicating factors in proper name generation. In addition, Reiter & Dale (2000) do not address how repeated references using a name in a text should be generated. For instance, should a discourse-old reference to our example-writer be realized as *Charles*, *Bukowski* or some combination of these and other attributes (e.g., using a modifier like *the poet Bukowski*)?

Imagine, for the sake of argument, that we would generate proper name references in a text by initially generating the full name, after which repeated references only consist of the last name (a.k.a. the family or surname). Intuitively, it is not difficult to come up with counterexamples to this "rule". Above we already discussed the difficulties of deciding what the most appropriate full name reference is for *Henry Charles Bukowski*, which (like *Keith Rupert Murdoch* and *Walter Bruce Willis*) seems to be the combination of middle and last names (as opposed to *Oprah Gail Winfrey* and *Serena Jameka Williams*, for who it is more common the combination of first and last names). Moreover, using the last name for repeated references may work well for the likes of *Winston Churchill* and *Angela Merkel*, but seems less suitable for *Napoleon Bonaparte* or *Madonna Ciccone*, to mention just two. Moreover, our example rule cannot account for the occurrence of modifiers. And, finally, it seems highly unlikely that human writers would adhere to such a strict rule. Rather, one might expect writers to vary in their choices of which name to use, depending on stylistic and discourse factors, much like the choice of referential form varies as a function of such factors (Castro Ferreira et al., 2016a,b).

In general, we know very little about how proper names should be generated in text – as far as we know, there have been hardly any systematic corpus studies and only very little concrete proposals on how to automatically generate proper name references. In this paper, we therefore present a large scale corpus analysis, and, based on this, two versions of a new probabilistic model of proper name generation: one that always chooses the most likely proper name form and one that relies on a 'roulettewheel' selection model and hence will generate more varied references. These models rely both on the nature of the entity referred to (what is the likelihood that a given person will be referred to using, say, the first or last

name?) and on the discourse context for generating proper name references in text. In an intrinsic evaluation experiment, we compare the performance of the two versions of this model with our implementations of the two proposals that have been made before (Siddharthan et al., 2011; van Deemter, 2016). We also describe a human evaluation experiment where we compare original texts with alternative versions that include proper names generated by our model.

5.2 Related work

Even though proper name references occur frequently in written text, their generation remains seriously understudied. A recent survey of REG models (Krahmer & van Deemter, 2012) has essentially nothing to say about the topic, and general surveys of automatic text generation such as Reiter & Dale (2000) only briefly mention a very basic rule (use a proper name, if available, for first references), without further specifying or evaluating it.

Recently, van Deemter (2016) has highlighted the importance of proper name generation. After discussing why a simple rule like the one proposed by Reiter and Dale cannot account for the complexities of proper name references in text, he argues that names could just be treated like other attributes in the generation of descriptions. Put differently, the name of an object can be modeled just like its color or size (typical attributes used in REG examples) – just as a description like *the tall man* rules out men that are not tall, so does a proper name like *Charles* rule out other people not named Charles. A standard REG algorithm, such as, for example, the Incremental Algorithm (Dale & Reiter, 1995) can then be used to compute when a name should be used and in which form. Van Deemter's work is

of a theoretical nature; he has not implemented or tested this idea, so we cannot tell how well it can account for proper name references in text. In addition, in this form, his proposal cannot account for possible variations in proper name form throughout a text.

The most detailed study of proper name generation, as far as we know, is the seminal study by Siddharthan et al. (2011), which (re-)generates references to people in news summaries. For their algorithm(s), the authors present two manually constructed rules, based on earlier theories of reference, one for discourse-new references (including the full name) and one for discourse-old references (which in full says: "Use surname only, remove all pre- and post-modifiers."). They discuss, based on corpus analyses, how notions like discourse-new and discourse-old can be learned without manual annotation, and how they co-determine whether additional attributes such as role and affiliation should be included. Finally, they show that their model leads to improved (more coherent) summaries. While the approach offers a very interesting solution for the generation of discourse-new proper name references with modifiers for major characters in a news story (*Former East German leader Erich Honecker*), the proper name generation rule itself is very similar to the example rule discussed in the introduction (use the full name for discourse-new references and only the surname for discourse-old references). It is not specified how the full name should be realized (remember the *Henry Charles Bukowski*-example), and neither can the approach deal with exceptions to the surname-only rule (remember the *Madonna Ciccone*-example) or with intratext variation.

5.3 REGnames

For our explorations, we relied on the REGnames corpus (Chapter 4; Castro Ferreira et al., 2016c). REGnames is a corpus of 53,102 proper names referring to 1,000 people in 15,241 texts. The corpus consists of webpages extracted from the Wikilinks corpus (Singh et al., 2012), which was initially collected for the study of cross-document coreference and consists of more than 40 million references to almost 3 million entities in around 11 million webpages. All the references annotated in Wikilinks were grouped according to the Wikipedia page of the entity. This procedure enables easy identification of the mentioned entity and facilitates the extraction of more information about it.

To build the REGnames corpus, Castro Ferreira et al. (2016c) (Chapter 4) selected the 1,000 most frequently mentioned people in the Wikilinks corpus. Then for each person, they selected random webpages from Wikilinks which mention the person at least once. On all selected webpages, part-of-speech tagging, lemmatization, named entity recognition, dependency parsing, syntactic parsing, sentiment analysis and coreference resolution was performed by using the Stanford CoreNLP software (Manning et al., 2014).

All extracted proper names were automatically annotated with their syntactic position (subject, object or genitive noun phrase in a sentence) and referential statuses in the text (discourse-new or discourse-old) and in the sentence (sentence-new or sentence-old). The extracted proper names were also annotated according to their form, i.e. which kind(s) of name (first, middle and/or last names), and modifier(s) (title and/or appositive) were part of the proper name. To check for the presence of first, middle and last names, a Proper Name Knowledge Base was extracted from DBpedia (Bizer et al., 2009) with all the names of the people in the corpus. Then, to

check for the presence of a title or an appositive, named entity recognition information and the dependency tree were used respectively.

In the corpus analysis, Castro Ferreira et al. (2016c) (Chapter 4) noticed that proper name references generally decrease in lengths across the text. They also concluded that a discourse-old or sentence-new proper name reference in the object position of a sentence tends to be shorter than a discourse-new or sentence-old proper name reference in the subject position of a sentence. In general, the corpus is a valuable resource which can be used to train a statistical model for proper name generation, as we show in the next section.

5.4 A model for proper name generation

Similarly to the generation of definite descriptions, our model produces a proper name reference in two sequential steps: content selection and linguistic realization.

5.4.1 Content selection

The content selection discussed here is analogous to the selection of semantic attributes (type, color, size, etc) when generating a description of an entity (Dale & Haddock, 1991; Dale & Reiter, 1995). However, instead of attributes, the content selection step in our model aims to choose the *form* of a proper name reference (which kind(s) of name and modifier(s) are part of the proper name reference).

Features By analyzing the REGnames corpus, Castro Ferreira et al. (2016c) (Chapter 4) observed that proper names vary in their forms throughout a text. Moreover, as discussed in the Introduction (Section 5.1), a

Feature	Description
Syntactic Position	Subject, object or a genitive noun phrase in the sentence.
Referential Status	First mention of the referent (new) or not (old) at the level of text and sentence.

Table 5.1: Discourse features that describe the references.

proper name form can also be influenced by the person to be mentioned. Thus, we conditioned the choice of a specific proper name form by a set of discourse features that describe the reference as well as to the person to be mentioned.

Table 5.1 depicts the discourse features used to describe the proper name references. We choose them based on the analysis of the REGnames corpus (Section 5.3).

Forms Our model selects a proper name form over all forms annotated on the REGnames corpus, i.e. a total of 28 possible ones. Table 5.2 depicts the most frequent ones. The complete list can be found at the webpage that describes the REGnames corpus[‡].

Notation Given a person p to be referred to by his/her proper name and the set of discourse features D that describe the reference, we aim to predict the form $f \in F$ of a proper name as Equation 5.1 shows.

$$P(f \mid D, p) = \frac{P(f \mid p) \prod_{d \in D} P(d \mid f, p)}{\sum_{f' \in F} P(f' \mid p) \prod_{d \in D} P(d \mid f', p)} \tag{5.1}$$

To account for unseen data, the conditional probabilities are computed

[‡] http://ilk.uvt.nl/~tcastrof/regnames/

using the additive smoothing technique with $\alpha = 1$. Equations 5.2 and 5.3 summarize the procedure.

$$P(f \mid p) = \frac{count(f \cap p) + \alpha}{count(p) + \alpha|F|} \tag{5.2}$$

$$P(d \mid f, p) = \frac{count(d \cap f \cap p) + \alpha}{count(f \cap p) + \alpha|D|} \tag{5.3}$$

Variation Besides the fact that proper name references may vary in their forms throughout a text and according to the person to be referred to, they may also vary in similar situations of a text. In an extrinsic evaluation comparing human- and machine-generated summaries, for instance, Siddharthan et al. (2011) reported that the lack of variation in the form of discourse-old proper names references was one of the disadvantages of their summarization system in the cases where human summaries were chosen. Our model fills this gap by performing Equation 5.1 over all the proper name forms given a set of similar references. That is proper name references to the same person and described by the same set of discourse feature values. This procedure results in a frequency distribution over all relevant proper name forms. Then, similar to the rouletewheel selection of Castro Ferreira et al. (2016b) (Chapter 3) for the choice of referential forms, we can randomly apply the frequencies into a group of similar references in such a way that their forms will be representative of the distribution predicted by the model. For instance, given a group of 5 references and a frequency distribution of 0.8 for the *first+last* form and 0.2 for the *last* form, 4 references would assume the first form, whereas 1 reference would assume the other one.

Form	Frequency
First+Last	46.2%
Last	34.9%
First	8.5%
Middle+Last	2.8%
First+Middle+Last	2.3%
Middle	1.5%
Others	3.8%

Table 5.2: Most popular proper name forms in REGnames corpus and their frequencies.

5.4.2 Linguistic realization

Once we select the form of a proper name reference to a person in a particular discourse context, we linguistically realize this reference by choosing the most likely words - including titles and proper nouns - to be part of it. The process is analogous to the linguistic realization of a set of attribute-values into a description (Bohnet, 2008; Zarriess & Kuhn, 2013). Equation 5.4 summarizes it.

$$P(n_1 \ldots n_t \mid f, p) = \prod_t P(n_t \mid n_{t-1}, \{e_i\}_{i=1}^{|f|}, p) \tag{5.4}$$

The vocabulary used in the linguistic realization step consists of all the titles found in REGnames, all the possible names of the given person present in the corpus' proper name knowledge base, and an *end* token, present at the end of all proper name references in the training set. The process finishes when this token is predicted ($n_t = END$). The choice of a word n_t is conditioned to the previous generated word in the proper name reference (n_{t-1}), the elements present in the given form ($\{e_i\}_{i=1}^{|f|}$: constrained to first, middle and last name; plus title and appositive) and

the person to be referred to (p). If $P(n_t \mid n_{t-1}, \{e_i\}_{i=1}^{|f|}, p) = 0$, we drop the less frequent element from the given proper name form. If all the elements were dropped and the probability would still be 0, we conditioned the choice only to the person ($P(n_t \mid p)$). Regarding the cases in which the original proper name form indicates the presence of an appositive, we add a description - obtained from Wikidata (Vrandečić & Krötzsch, 2014) - at the end of the generated proper name reference.

5.5 Baselines

In order to evaluate the performance of our model, we developed three baseline models. All the models have their outputs constrained to three choices: given name, surname and full name of a person. Given name and surname are determined by the values of the following attributes in the person's DBpedia page: *foaf:givenName* and *foaf:surname*. Full name was defined as the combination of both values. If these attributes are missing, we use the birth name of the person, also extracted from DBpedia (*dbp:birthName*). In this situation, the full name of a person will be the proper birth name, whereas given and surnames will be the first and last tokens from the birth name, respectively.

The first baseline, called *Random*, is a baseline that randomly chooses one of the three options to generate a proper name.

The second baseline is an adaptation of the model proposed by van Deemter (2016) and will be called *Deemter*. Among the full name, given name and surname of a person, our adaptation chooses the shortest name that distinguishes the mentioned person from all other entities in the current and previous 3 sentences in the text. It is important to stress that this model is our adaptation, since the proposal of van Deemter (2016) only

applies for initial references, not for repeated ones in a text.

Finally, the third system we compare against is based on Siddharthan et al. (2011) and will be called *Siddharthan*. This baseline chooses the full name of a person for discourse-new references; and his/her surname otherwise.

5.6 Automatic evaluation

We intrinsically evaluate the models by training and testing them on a subset of the REGnames corpus. This evaluation aims to investigate how close our model can produce proper name references to the ones generated by human writers.

5.6.1 Data

We considered a subset of the REGnames corpus as our evaluation data. From the 1,000 people in the corpus, we first filtered the ones whose birth names were not mentioned, or for whom the values of the DBpedia's attributes *foaf:name*, *foaf:givenName* and *foaf:surname* were missing. This measure was taken in order to have a consistent vocabulary to linguistically realize the proper name references, as well as to make sure that our baselines would always have a consistent output. Then, from the remaining people, we only selected the ones with at least 50 proper name references in the REGnames corpus such that we could train and test our model properly. In total, we used 43,655 proper names references to 432 people as our evaluation data.

In order to investigate the influence of the text domain in the generation of proper names, we classified the webpages from where our evaluation data were extracted according to 3 domains: Blog, News and Wiki.

All the webpages whose the url contained the substrings *blog*, *tumblr* or *wordpress* were classified as part of the blog domain. If the substrings were *new* or *article*, the webpage was classified as a news. Finally, we classified as Wiki all the webpages whose the url contained the substring *wiki*. All the other webpages were grouped into a *Other domains* category.

5.6.2 Method

10-fold-cross-validation was performed to evaluate the models. We made sure that the number of references per person was uniform among the folds. To measure the models performance in the choice of the proper name form, accuracy was used. To check the similarity among the realized proper name reference and the gold standard one, we used the string edit distance.

5.6.3 Models

We evaluated the three proposed baselines (*Random*, *Deemter* and *Siddharthan*) and two versions of our model: *PN-Variation* and *PN+Variation*.

PN-Variation does not take the variation into account in the content selection. In other words, this model always chooses the most likely proper name form for the references in the test set which refer to the same person and are described by the same combination of discourse feature values. On the other hand, *PN+Variation* takes variation into account by applying the distribution of proper name forms obtained from the training set to the similar references in the test set, as explained in Section 5.4.1.

5.6.4 Results

Table 5.3 summarizes the accuracy-scores of the models in the prediction of the proper name forms. Both versions of our model outperform the

Model	Blog	News	Wiki	Other domains	Overall
Random	0.25	0.22	0.22	0.25	0.25
Deemter	0.33	0.30	0.28	0.33	0.33
Siddharthan	0.52	0.48	0.42	0.45	0.48
PN-Variation	0.66	0.63	0.66	0.70	**0.68**
PN+Variation	0.58	0.55	0.59	0.63	0.60

Table 5.3: Proper name form accuracies of our two models (PN-Variation and PN+Variation) as a function of text genre and compared to three baseline models (Random, Deemter, Siddharthan).

baselines for all the domains. *PN-Variation* is the model with the highest accuracy.

Figure 5.1 depicts the string edit distance among the gold standard proper names and the ones generated by the proposed models. A Repeated Measures ANOVA determined that the string edit distances of the models were significantly different ($F(4, 36) = 1630, p < .001$). We performed a post hoc analysis with paired t-test using Bonferroni adjusted alpha levels of 0.005 per test ($0.05/10$). Both versions of our model significantly outperformed the baselines with all pairwise comparisons significant at $p < .001$. Regarding the comparison of our models, *PN-Variation* is significantly better than *PN+Variation* ($t(9) = -38.14, p < .001$).

Figure 5.2 shows the evaluation of our models by domain. A Repeated Measures ANOVA shows that the string edit distances of the models were significantly different in all domains (Blog: $F(4, 36) = 718.8, p < .001$; News: $F(4, 36) = 308.2, p < .001$; Wiki: $F(4, 36) = 118.5, p < .001$; Other domains: $F(4, 36) = 2213, p < .001$).

We also performed a post hoc analysis for the results by domain in the same style we did for the general results. In the blog and news domains, both versions of our model significantly outperformed all the base-

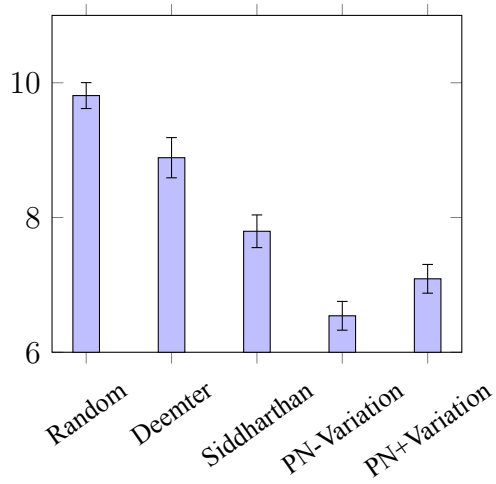


Figure 5.1: String edit distance in the overall corpus. Error bars represent 95% confidence intervals.

lines with all pairwise comparisons significant at $p < .005$. Among our models, *PN-Variation* is significantly better than *PN+Variation* (Blog: $t(9) = -26.33, p < .001$; News: $t(9) = -7.45, p < .001$).

In the wiki domain and in texts which are not part of the blog, news and wiki domain, both versions of our model also significantly outperformed all the baselines with all pairwise comparisons significant at $p < .001$. The difference in the results of *PN-Variation* and *PN+Variation* is also significant (Wiki: $t(9) = -4.91, p < .001$; Other domains: $t(9) = -27.14, p < .001$)

5.7 Human evaluation

We also performed a human evaluation aiming to compare original texts with alternative versions whose proper name references were generated by our model. This evaluation aims to investigate the quality of the proper

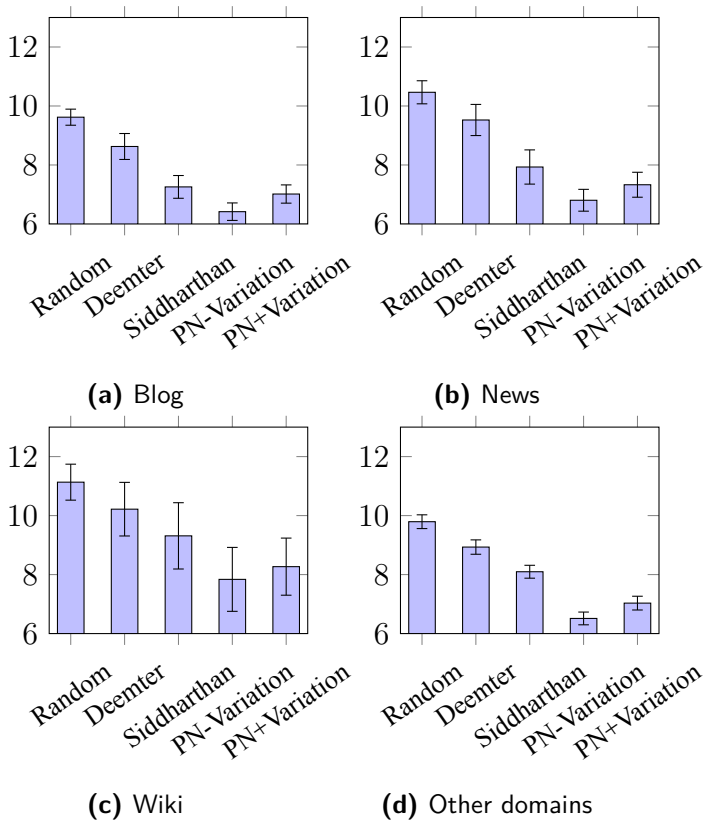


Figure 5.2: String edit distances of the models in the (5.2a) blog, (5.2b) news, (5.2c) wiki and (5.2d) in other domains which are not the previous ones. Error bars represent 95% confidence intervals.

name references from the perspective of the human reader.

5.7.1 Materials

We used 9 abstracts from English Wikipedia pages whose topic is one of the people studied in the REGnames corpus. They were extracted from DBpedia and have at least 10 proper name references to the topic.

Although our model did not yield its best results for this domain, it was chosen based on the relatively short length of the texts and the large amount of proper name references they have. Moreover, the proper name references in Wikipedia abstracts are similar to the ones generated by our *Siddharthan* baseline, i.e. a full name to discourse-new people, and surname to discourse-old people.

5.7.2 Method

For each abstract, we designed 3 trials. In the first, we presented participants with the original text next to the version with the proper name references generated by the *PN-Variation* model (Original vs. No Variation). In the second, we presented the original text next to the version with the proper name references generated by the *PN+Variation* (Original vs. Variation). Finally, the third trial consists of the text versions with the proper name references produced by both versions of our model (No Variation vs. Variation). The trials of a text were distributed in different lists such that we obtained 3 lists with 9 texts - 3 trials of each type in a list. In all the texts, the proper name references were highlighted in yellow. For each trial, we asked participants to choose which text they preferred, taking into account the highlighted references. The experiment is publicly

available[§].

We recruited 60 participants through Crowdflower – 20 per list. Of the participants, 44 were female and their average age was 36 years. All participants reported to be proficient in the English language (58 were native speakers).

5.7.3 Results

The texts of the "Original" version were the favorite of 69% of the participants in comparison with texts of the "No Variation" version (Chi-square $\chi^2(2) = 25.69, p < .001$), and 75% participants with the "Variation" version (Chi-square $\chi^2(2) = 45; p < .001$). Regarding the "No Variation vs. Variation" trials, texts of the "No Variation" version were the favorite of the participants in 59% of the cases (Chi-square $\chi^2(2) = 6.42; p < .05$).

5.8 General discussion

Proper name generation is a seriously understudied phenomenon in automatic text generation. There are many different ways in which a person can be referred to in a text using their name (*Barack Hussein Obama II*, *Barack Obama*, *Obama*, *President Obama*, etc.) and arguably a text that uses different naming formats in different conditions is more human-like than one that relies on a fixed strategy (e.g., always use the full name).

This paper introduced a new statistical model for the generation of proper names in text, taking into account three different factors: (1) who the person is, (2) in which discourse context the proper name reference should be generated and (3) the different forms that a proper name can assume in similar situations (variation). The model was developed based

[§] http://ilk.uvt.nl/~tcastrof/eacl2017

on the REGnames corpus (Chapter 4; Castro Ferreira et al., 2016c), which contains a large number of proper name references in various discourse situations. We also implemented two other systems for the sake of comparison: one based on the Siddharthan et al. (2011) model and one based on the ideas for proper name reference proposed by van Deemter (2016).

We developed two versions of our model: one that deterministically generated the best proper name form in a given setting (*PN-variation*), and one that relied on a probabilistic distribution over different forms, allowing for more variation in the output (*PN+Variation*). Both models were systematically compared to a random baseline and the two alternative models due to Siddharthan et al. (2011) and van Deemter (2016).

Automatic Evaluation We first conducted an automatic evaluation investigating to what extent the evaluated models produced proper name references similar to the ones generated by human writers, using a held-out subset of the REGnames corpus. In general, we found that both versions of our model were able to outperform a random baseline and the two reference systems, where the version without variation (*PN-Variation*) yielded the best results. Across text domains, there was variation in the performance of both versions of our model. The worst results were registered in the Wiki domain, suggesting that text domain is a factor that may be taken into account in the task of generating proper names.

Human Evaluation In the automatic evaluation experiment, the differences between the system with and without variation were small, so in a second study we asked whether human readers preferred the output from one of these systems over the other. For this purpose, we conducted an experiment consisting of pairwise comparisons based on texts taken from the Wikipedia domain, where we compared the output produced by the

PN-variation and the *PN+variation* system with the original text and also among them. Interestingly, we found that people had a general preference for the no-variation model over the one that non-deterministically generated varied texts. This suggests that readers prefer consistency in proper name references to the same topic in similar situations, which is different from the choice of referential *form* (Chapter 3; Castro Ferreira et al., 2016b).

Additionally, we found that participants preferred the original over the regenerated texts. We suspect that this preference was due to the initial discourse-new proper name reference, which in the Wikipedia texts has a special status. Usually, the initial reference to the topic is not the most common proper name reference in other domains, but a specific Wikipedia format which our system does not produce. For example, the original text about Magic Johnson starts with *Earvin "Magic" Johnson Jr.* in the discourse-new proper name reference, while our system simply produced *Magic Johnson*.

Semantic web Earlier work on REG models has concentrated on the generation of descriptions, typically assuming the existence of a knowledge base of entities (Dale & Haddock, 1991; Dale & Reiter, 1995) or introducing one to small domains (Gatt & Belz, 2010). Our REG models for proper names, however, strongly rely on the semantic web as an information resource of the entities to be referred to. Databases like DBpedia (Bizer et al., 2009) and Wikidata (Vrandečić & Krötzsch, 2014) provide information about thousands of entities and can be used in different domains.

Baselines We developed two powerful baselines based on proposals that have been made before. *Deemter* (van Deemter, 2016) relies on the cri-

teria of the first developed REG models (Dale & Haddock, 1991; Dale & Reiter, 1995): given a target, produce a reference that distinguishes it from the distractors in the context. Our model as presented does not make this assumption (it does not always produce a proper name reference that distinguishes the target from the distractors). However, this could be incorporated into our model as well. For instance, given a list of the most likely proper name references produced by our model in a situation, we can choose the one with the highest likelihood that distinguishes the target from all other entities in the current and previous 3 sentences in the text (as in the *Deemter* model).

Regarding performance, *Siddharthan* is the baseline that performed best. The original version, proposed in Siddharthan et al. (2011), is even able to decide whether to include a modifier in a discourse-new reference based on the global salience of the entity mentioned. However, the model is arguably more limited in the production of a proper name itself. By always generating a surname in discourse-old references for instance, the Siddharthan model is not able to generate at least 10% of the references in the REGnames corpus (8.5% consist of *first name* references, and 1.5% of *middle name* ones).

Conclusion In sum, we conclude that our model is able to generate proper name references similar to the ones produced by human writers. In future research, it would be interesting to further investigate the role of text genre in proper name references as well as the influence of variation on proper name forms.

6

NeuralREG: An end-to-end approach to referring expression generation

Abstract Traditionally, Referring Expression Generation (REG) models first decide on the form and then on the content of references to discourse entities in text, typically relying on features such as salience and grammatical function. In this chapter, we present a new approach (NeuralREG), relying on deep neural networks, which makes decisions about form and content in one go without explicit feature extraction. Using a delexicalized version of the WebNLG corpus, we show that the neural model substantially improves over two strong baselines. Data and models are publicly available*.

*https://github.com/ThiagoCF05/NeuralREG

This chapter is based on Castro Ferreira, T., Moussallem, D., Kádár, Á., Wubben, S. & Krahmer, E. (2018). NeuralREG: An end-to-end approach to referring expression generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, ACL'2018 (pp. 1959–1969). Melbourne, Australia: Association for Computational Linguistics.

6.1 Introduction

Natural Language Generation (NLG) can be characterized as the task of automatically converting non-linguistic data into coherent natural language text (Reiter & Dale, 2000; Gatt & Krahmer, 2018). Since the input data will often consist of entities and the relations between them, generating references for these entities is a core task in many NLG systems (Dale & Reiter, 1995; Krahmer & van Deemter, 2012). Referring Expression Generation (REG), the task responsible for generating these references, is typically presented as a two-step procedure. First, the referential form needs to be decided, asking whether a reference at a given point in the text should assume the form of, for example, a proper name ("Frida Kahlo"), a pronoun ("she") or description ("the Mexican painter"). In addition, the REG model must account for the different ways in which a particular referential form can be realized. For example, both "Frida" and "Kahlo" are name-variants that may occur in a text, and she can alternatively also be described as, say, "the magical realist" or "the famous female painter".

Most of the earlier approaches to REG focuses either on selecting referential form (Orita et al., 2015; Castro Ferreira et al., 2016b), or on selecting referential content, typically zooming in on one specific kind of reference such as a pronoun (e.g., Henschel et al., 2000; Callaway & Lester, 2002), definite description (e.g., Dale & Haddock, 1991; Dale & Reiter, 1995) or proper name generation (e.g., Siddharthan et al., 2011; van Deemter, 2016; Castro Ferreira et al., 2017b). Instead, in this paper, we propose NeuralREG: an end-to-end approach addressing the full REG task, which given a number of entities in a text, produces corresponding referring expressions, simultaneously selecting both form and content. Our approach is based on neural networks which generate referring expressions to discourse entities relying on the surrounding linguistic context, without the

use of any feature extraction technique.

REG models can be used in different ways in NLG systems, ranging from a dedicated task in traditional pipeline models (Reiter & Dale, 2000) or as a way to fill gaps in syntactic templates (e.g., Theune et al., 2001). Interestingly, REG also becomes relevant in modern "end-to-end" NLG approaches, which model the entire generation process in an integrated manner (see e.g. Konstas et al., 2017; Gardent et al., 2017b). Recently, in order to decrease data sparsity, these models work on inputs where references to entities have been replaced for general tags (ENTITY-1, ENTITY-2, etc.) according to a process known as *Delexicalization*. Based on the delexicalized input, the model generates outputs which may be likened to templates in which references to the discourse entities are not realized (as in "The ground of ENTITY-1 is located in ENTITY-2.").

While our approach is compatible with different applications of REG models, in this paper we concentrate on the last application, relying on a specifically constructed set of 78,901 referring expressions to 1,501 entities in the context of the semantic web, derived from a (delexicalized) version of the WebNLG corpus (Gardent et al., 2017a,b). Both this data set and the model are publicly available. We compare NeuralREG against two baselines in an automatic and human evaluation, showing that the integrated neural model is a marked improvement.

6.2 Related work

In recent years, we have seen a surge of interest in using (deep) neural networks for a wide range of NLG-related tasks. This includes, for example, the generation of (first sentences of) Wikipedia entries (e.g., Lebret et al., 2016), the generation of poetry (e.g., Zhang & Lapata, 2014), and the gen-

Subject	Predicate	Object
108_St_Georges_Terrace	location	Perth
Perth	country	Australia
108_St_Georges_Terrace	completionDate	1988*@year*
108_St_Georges_Terrace	cost	120 million (Australian dollars)*@USD*
108_St_Georges_Terrace	floorCount	50*@Integer*

↓

108 St Georges Terrace was completed in 1988 in Perth, Australia. It has a total of 50 floors and cost 120m Australian dollars.

Figure 6.1: Example of a set of triples (top) and corresponding text (bottom).

eration of text from abstract meaning representations (e.g., Konstas et al., 2017; Castro Ferreira et al., 2017a). Also applications for controlling the style of generated outputs (e.g., Ficler & Goldberg, 2017) and the generation of image descriptions (e.g., Bernardi et al., 2016) have generated a lot of interest. So far, the usage of deep neural networks for REG has remained limited, however, and we are not aware of any other integrated, end-to-end model for generating referring expressions in text.

There is, however, a lot of earlier work on selecting the form and content of referring expressions, both in psycholinguistics and in computational linguistics. In psycholinguistic models of reference, various linguistic factors have been proposed as influencing the form of referential expressions, including cognitive status (Gundel et al., 1993) and centering (Grosz et al., 1995), but also information density (Jaeger, 2010). In models such as these, notions like *salience* and *accessibility* play a central role, where it is assumed that entities which are salient in the discourse are more likely to be referred to using shorter referring expressions (like pronouns) than less salient entities, which are typically referred to using

longer expressions (like full proper names).

These models also have to account for what causes a referent to be salient in a discourse context. Brennan (1995), for example, pointed out that grammatical role is a factor in this, arguing that a referent in subject position of a sentence is more salient than a referent in object position. Chafe (1994) made a case for givenness and recency, pointing out that a subsequent reference relatively close to its antecedent is more salient than one which is further away. Arnold (1998), to give a last example, discussed topicality and parallelism, claiming that topical referents in parallel syntactic positions to their antecedent references tend to be more salient than non-topic referents which are not in parallel.

Building on these ideas, many REG models for generating references in texts also strongly rely on the concept of salience and factors contributing to it. Reiter & Dale (2000) for instance, discussed a straightforward rule-based method based on this notion, stating that "in a domain where the entities have proper names, we might choose to always use a full proper name for initial reference, perhaps with an appositive noun phrase to indicate properties that are deemed relevant in the domain of application". For subsequent references, they propose the use of a pronoun in case there is no mention to any other entity of same person, gender and number between the reference and its antecedents. Otherwise, subsequent references may be realized with a description. This normative approach makes intuitive sense, but does not capture the rich variety that can be found in actual text. More recently, Castro Ferreira et al. (2016b) (Chapter 3) proposed a data-driven, non-deterministic model for generating referential forms, taking into account salience features extracted from the discourse such as grammatical position, givenness and recency of the reference. Importantly, these models do not specify which contents a particular reference,

be it a proper name or description, should have. For this, separate models are typically used, including, for example, Dale & Reiter (1995) for generating descriptions, Siddharthan et al. (2011); van Deemter (2016) for proper names, and many others.

Of course, when texts are generated in practical settings, both form and content need to be chosen. This was the case, for instance, in the GREC shared task (Belz et al., 2010), which aimed to evaluate models for automatically generated referring expressions grounded in discourse. In one of the tasks, the input for the models were delexicalized Wikipedia articles, i.e., texts in which the referring expressions to the topic of the relevant Wikipedia entry were removed and appropriate references throughout the text needed to be generated (by selecting, for each gap, from a list of candidate referring expressions of different forms and with different contents). Some participating systems approached this with traditional pipelines for selecting referential form, followed by referential content, while others proposed more integrated methods. For example, Gupta & Bandopadhyay (2009) presented a combination of rules and machine learning techniques based on features such as semantic and syntactic category, paragraph and sentence positions, and reference number. Similarly, Greenbacker et al. (2010) proposed a decision tree that also made use of recency and part-of-speech features to choose a referring expression in a given discourse context.

In sum, REG models for text generation strongly rely on abstract features such as the salience of a referent for deciding on the form or content of a referent. Typically, these features are extracted automatically from the context, and engineering relevant features can be complex (as we have seen, different proposals on what constitutes salience have been put forward). Many of these models only address part of the problem, ei-

ther concentrating on the choice of referential form or on deciding on the contents of, for example, proper names or definite descriptions. The few end-to-end approaches usually work by computing salience features in order to generate referring expressions, for example, in the GREC context. In contrast, we introduce NeuralREG, an end-to-end approach based on neural networks which generates referring expressions to discourse entities directly from a delexicalized/wikified text fragment, without the use of any feature extraction technique. Below we describe our model in more detail, as well as the data on which we develop and evaluate it.

6.3 Data and processing

6.3.1 WebNLG corpus

Our data is based on the WebNLG corpus (Gardent et al., 2017a), which is a parallel resource initially released for the eponymous NLG challenge. In this challenge, participants had to automatically convert non-linguistic data from the Semantic Web into a textual format (Gardent et al., 2017b). The source side of the corpus are sets of *Resource Description Framework* (RDF) triples, the best known protocol used on the Semantic Web. Each RDF triple is formed by a Subject, Predicate and Object, where the Subject and Object are constants or Wikipedia entities, and predicates represent a relation between these two elements in the triple. The target side contains English texts, obtained by *crowdsourcing*, which describe the source triples. Figure 6.1 depicts an example of a set of 5 RDF triples and the corresponding text.

According to Gardent et al. (2017b), the corpus consists of 25,298 texts describing 9,674 sets of up to 7 RDF triples (an average of 2.62 texts per set) in 15 domains (Astronaut, University, Monument, Building, Comics

Tag	Entity
AGENT-1	108_St_Georges_Terrace
BRIDGE-1	Perth
PATIENT-1	Australia
PATIENT-2	1988*@year*
PATIENT-3	"120 million (Australian dollars)"*@USD*
PATIENT-4	50*@Integer*

AGENT-1 was completed in **PATIENT-2** in **BRIDGE-1** , **PATIENT-1** .
AGENT-1 has a total of **PATIENT-4** floors and cost **PATIENT-3** .

$\downarrow_{Wiki}$

108_St_Georges_Terrace was completed in **1988** in **Perth** , **Australia** .
108_St_Georges_Terrace has a total of **50** floors and cost
20_million_(Australian_dollars) .

Figure 6.2: Mapping between tags and entities for the related delexicalized/wikified templates.

Character, Food, Airport, Sports Team, Written Work, City, Athlete, Artist, Mean of Transportation, Celestial Body and Politician). In order to be able to train and evaluate our models for referring expression generation (the topic of this chapter), we produced a delexicalized version of the original corpus.

6.3.2 Delexicalized WebNLG

We delexicalized the training and development parts of the WebNLG corpus by first automatically mapping each entity in the source representation to a general tag. All entities that appear on the left and right side of the triples were mapped to AGENTs and PATIENTs, respectively. If an enti-

tiy appears on both sides of a relation, it cannot be mapped to either the agent or patient role, so these entities are represented as BRIDGEs. To distinguish different AGENTs, PATIENTs and BRIDGEs in a set, an ID is given to each entity of each kind (PATIENT-1, PATIENT-2, etc.). Once all entities in the text were mapped to different roles, the first two authors of this chapter manually replaced the referring expressions in the original target texts by their respective tags. Figure 6.2 shows the entity mapping and the delexicalized template for the example in Figure 6.1 in its versions with general tags and Wikipedia IDs to represent the references.

We delexicalized 20,198 distinct texts describing 7,812 distinct sets of RDF triples, resulting in 16,628 distinct templates. While this dataset (which we make available) has various uses, we used it to extract a collection of referring expressions to Wikipedia entities in order to evaluate how well our REG model can produce references to entities throughout a (small) text.

6.3.3 Referring expression collection

Using the delexicalized version of the WebNLG corpus, we automatically extracted all referring expressions by tokenizing the original and delexicalized versions of the texts and then finding the non overlapping items. For instance, by processing the text in Figure 6.1 and its delexicalized version in Figure 6.2, we would extract referring expressions like "108 St Georges Terrace" to ⟨ *AGENT-1, 108_St_Georges_Terrace* ⟩, "Perth" to ⟨ *BRIDGE-1, Perth* ⟩, "Australia" to ⟨ *PATIENT-1, Australia* ⟩ and so on.

Once all texts were processed and the referring expressions extracted, we filtered only the ones referring to Wikipedia entities, removing references to constants like dates and numbers, for which no references are generated by the model. In total, the final version of our dataset contains

78,901 referring expressions to 1,501 Wikipedia entities. To have a training, development and test split of the corpus, we used the referring expressions in 10% of the texts randomly chosen from the training set of the original WebNLG corpus as our development set, whereas the referring expression of the remaining texts were used as our training set. As test set, we used the referring expressions from the texts that originally were from the development set in WebNLG corpus. In total, we have 63,061, 7,097 and 8,743 referring expressions in the training, development and test sets, respectively.

Each instance of the final dataset consists of a truecased tokenized referring expression, the target entity (distinguished by its Wikipedia ID), and the discourse context preceding and following the relevant reference (we refer to these as the pre- and pos-context). Pre- and pos-contexts are the lowercased, tokenized and delexicalized pieces of text before and after the target reference. References to other discourse entities in the pre- and pos-contexts are represented by their Wikipedia ID, whereas constants (numbers, dates) are represented by a one-word ID removing quotes and replacing white spaces with underscores (e.g., *120_million_(Australian_dollars)* for "120 million (Australian dollars)" in Figure 6.2).

Although the references to discourse entities are represented by general tags in a delexicalized template produced in the generation process (AGENT-1, BRIDGE-1, etc.), for the purpose of disambiguation, NeuralREG's inputs have the references represented by the Wikipedia ID of their entities. In this context, it is important to observe that the conversion of the general tags to the Wikipedia IDs can be done in constant time during the generation process, since their mapping, like the first representation in Figure 6.2, is the first step of the generation process. In the next section,

we show in detail how NeuralREG models the problem of generating a referring expression to a discourse entity. In the next section, we show how NeuralREG models the problem of generating a referring expression to a discourse entity.

6.4 NeuralREG

NeuralREG aims to generate a referring expression $y = \{y_1, y_2, \cdots, y_T\}$ with T tokens to refer to a target entity token $x^{(wiki)}$ given a discourse pre-context $X^{(pre)} = \{x_1^{(pre)}, x_2^{(pre)}, \cdots, x_m^{(pre)}\}$ and pos-context $X^{(pos)} = \{x_1^{(pos)}, x_2^{(pos)}, \cdots, x_l^{(pos)}\}$ with m and l tokens, respectively. Equation 6.1 depicts the process.

$$P(y \mid X^{(pre)}, x^{(wiki)}, X^{(pos)}) \tag{6.1}$$

NeuralREG is implemented as a multi-encoder, attention-decoder network with bidirectional (Schuster & Paliwal, 1997) Long-Short Term Memory Layers (LSTMs) (Hochreiter & Schmidhuber, 1997) sharing the same input word-embedding matrix V.

6.4.1 Context encoders

Our model starts by encoding the pre- and pos-contexts with two separate bidirectional (Schuster & Paliwal, 1997) LSTM encoders (Hochreiter & Schmidhuber, 1997). These modules learn feature representations of the text surrounding the target entity $x^{(wiki)}$, which are used for the referring expression generation. The pre-context $X^{(pre)} = \{x_1^{(pre)}, x_2^{(pre)}, \cdots, x_m^{(pre)}\}$ is represented by forward and backward hidden-state vectors $(\overrightarrow{h}_1^{(pre)}, \cdots, \overrightarrow{h}_m^{(pre)})$ and $(\overleftarrow{h}_1^{(pre)}, \cdots, \overleftarrow{h}_m^{(pre)})$. The final annotation vector for each encoding timestep t is obtained by the concatena-

tion of the forward and backward representations $h_t^{(pre)} = [\overrightarrow{h}_t^{(pre)}, \overleftarrow{h}_t^{(pre)}]$. The same process is repeated for the pos-context resulting in representations $(\overrightarrow{h}_1^{(pos)}, \cdots, \overrightarrow{h}_l^{(pos)})$ and $(\overleftarrow{h}_1^{(pos)}, \cdots, \overleftarrow{h}_l^{(pos)})$ as well as annotation vectors $h_t^{(pos)} = [\overrightarrow{h}_t^{(pos)}, \overleftarrow{h}_t^{(pos)}]$. Finally, the encoding of target entity $x^{(wiki)}$ is simply its entry in the shared input word-embedding matrix V_{wiki}.

6.4.2 Decoder

The referring expression generation module is an LSTM decoder implemented in 3 different versions: `Seq2Seq`, `CAtt` and `HierAtt`. All decoders at each timestep i of the generation process take as input features their previous state s_{i-1}, the target entity-embedding V_{wiki}, the embedding of the previous word of the referring expression $V_{y_{i-1}}$ and finally the summary vector of the pre- and pos-contexts c_i. The difference between the decoder variations is the method to compute c_i.

`Seq2Seq` models the context vector c_i at each timestep i concatenating the pre- and pos-context annotation vectors averaged over time:

$$\hat{h}^{(pre)} = \frac{1}{N}\sum_{i}^{N} h_i^{(pre)} \tag{6.2}$$

$$\hat{h}^{(pos)} = \frac{1}{N}\sum_{i}^{N} h_i^{(pos)} \tag{6.3}$$

$$c_i = [\hat{h}^{(pre)}, \hat{h}^{(pos)}] \tag{6.4}$$

`CAtt` is an LSTM decoder augmented with an attention mechanism (Bahdanau et al., 2015) over the pre- and pos-context encodings, which is used to compute c_i at each timestep. We compute energies $e_{ij}^{(pre)}$ and $e_{ij}^{(pos)}$ be-

tween encoder states $h_i^{(pre)}$ and $h_i^{(post)}$ and decoder state s_{i-1}. These scores are normalized through the application of the softmax function to obtain the final attention probability $\alpha_{ij}^{(pre)}$ and $\alpha_{ij}^{(post)}$. Equations 6.5 and 6.6 summarize the process with k ranging over the two encoders ($k \in [pre, pos]$), being the projection matrices $W_a^{(k)}$ and $U_a^{(k)}$ as well as attention vectors $v_a^{(k)}$ trained parameters.

$$e_{ij}^{(k)} = v_a^{(k)T} \tanh(W_a^{(k)} s_{i-1} + U_a^{(k)} h_j^{(k)}) \tag{6.5}$$

$$\alpha_{ij}^{(k)} = \frac{\exp(e_{ij}^{(k)})}{\sum_{n=1}^{N} \exp(e_{in}^{(k)})} \tag{6.6}$$

In general, the attention probability $\alpha_{ij}^{(k)}$ determines the amount of contribution of the jth token of the k-context in the generation of the ith token of the referring expression. In each decoding step i, a final summary-vector for each context $c_i^{(k)}$ is computed by summing the encoder states $h_j^{(k)}$ weighted by the attention probabilities $\alpha_i^{(k)}$:

$$c_i^{(k)} = \sum_{j=1}^{N} \alpha_{ij}^{(k)} h_j^{(k)} \tag{6.7}$$

To combine $c_i^{(pre)}$ and $c_i^{(pos)}$ into a single representation, this model simply concatenate the pre- and pos-context summary vectors $c_i = [c_i^{(pre)}, c_i^{(pos)}]$.

`HierAtt` implements a second attention mechanism inspired by Libovický & Helcl (2017) in order to generate attention weights for the pre- and pos-context summary-vectors $c_i^{(pre)}$ and $c_i^{(pos)}$ instead of concatenate them. Equations 6.8, 6.9 and 6.10 depict the process, being the projection matrices $W_b^{(pre)}$, $W_b^{(pos)}$, $U_b^{(pre)}$ and $U_b^{(pos)}$ as well as attention vectors

$v_b^{(pre)}$ and $v_b^{(pos)}$ trained parameters.

$$e_i^{(k)} = v_b^{(k)T} \tanh(W_b^{(k)} s_{i-1} + U_b^{(k)} c_i^{(k)}) \tag{6.8}$$

$$\beta_i^{(k)} = \frac{\exp(e_i^{(k)})}{\sum_n \exp(e_i^{(n)})} \tag{6.9}$$

$$c_i = \sum_k \beta_i^{(k)} U_b^{(k)} c_i^{(k)} \tag{6.10}$$

Decoding Given the summary-vector c_i, the embedding of the previous referring expression token $V_{y_{i-1}}$, the previous decoder state s_{i-1} and the entity-embedding V_{wiki}, the decoders predict their next state which later is used to compute a probability distribution over the tokens in the output vocabulary for the next timestep as Equations 6.11 and 6.12 show.

$$s_i = \Phi_{\text{dec}}(s_{i-1}, [c_i, V_{y_{i-1}}, V_{wiki}]) \tag{6.11}$$

$$\begin{aligned} p(y_i | y_{<i}, X^{(pre)}, x^{(wiki)}, X^{(pos)}) = \\ \text{softmax}(W_c s_i + b) \end{aligned} \tag{6.12}$$

In Equation 6.11, s_0 and c_0 are zero-initialized vectors. In order to find the referring expression y that maximizes the likelihood in Equation 6.12, we apply a beam search with length normalization with $\alpha = 0.6$ (Wu et al., 2016):

$$lp(y) = \frac{(5 + |y|)^\alpha}{(5 + 1)^\alpha} \tag{6.13}$$

The decoder is trained to minimize the negative log likelihood of the next token in the target referring expression:

$$J(\theta) = -\sum_{i} \log \mathrm{p}(y_i | y_{<i}, X^{(pre)}, x^{(wiki)}, X^{(pos)}) \tag{6.14}$$

6.5 Models for comparison

We compared the performance of NeuralREG against two baselines: *OnlyNames* and a model based on the choice of referential form method of Castro Ferreira et al. (2016b) (Chapter 3), dubbed *Ferreira*.

OnlyNames is motivated by the similarity among the Wikipedia ID of an element and a proper name reference to it. This method refers to each entity by their Wikipedia ID, replacing each underscore in the ID for whitespaces (e.g., *Appleton_International_Airport* to *"Appleton International Airport"*).

Ferreira works by first choosing whether a reference should be a proper name, pronoun, description or demonstrative. The choice is made by a Naive Bayes method as Equation 6.15 depicts.

$$P(f \mid X) \propto \frac{P(f) \prod_{x \in X} P(x \mid f)}{\sum_{f' \in F} P(f') \prod_{x \in X} P(x \mid f')} \tag{6.15}$$

The method calculates the likelihood of each referential form f given a set of features X, consisting of grammatical position and information status (new or given in the text and sentence). Once the choice of referential form is made, the most frequent variant is chosen in the training corpus

given the referent, syntactic position and information status. In case a referring expression for a wiki target is not found in this way, a back-off method is applied by removing one factor at a time in the following order: sentence information status, text information status and grammatical position. Finally, if a referring expression is not found in the training set for a given entity, the same method as *OnlyNames* is used. Regarding the features, syntactic position distinguishes whether a reference is the subject, object or subject determiner (genitive) in a sentence. Text and sentence information statuses mark whether a reference is a initial or a subsequent mention to an entity in the text and the sentence, respectively. All features were extracted automatically from the texts using the sentence tokenizer and dependency parser of Stanford CoreNLP (Manning et al., 2014).

6.6 Automatic evaluation

6.6.1 Data

We evaluate our models on the training, development and test referring expression sets described in Section 6.3.3.

6.6.2 Metrics

We compare the referring expressions produced by the evaluated models with the gold-standards ones using accuracy and String Edit Distance (Levenshtein, 1966). Since pronouns are highlighted as the most likely referential form to be used when a referent is salient in the discourse, as argued in the introduction, we also computed pronoun accuracy, precision, recall and F1-score in order to evaluate how well the models capture discourse salience. Finally, as a measure of coherence, we lexicalized the original

templates with the referring expressions produced by the models and compare them with the original texts in the corpus using BLEU score (Papineni et al., 2002). Since our model did not generate referring expressions for constants (dates and numbers), we just copied their source version into the template. Note that because only the referring expressions are potentially different from the original texts, BLEU scores will be highly. Crucially, however, differences in BLEU scores between different systems are only attributable to the generated references.

Post hoc McNemar's and Wilcoxon signed ranked tests adjusted by the Bonferroni method were used to test the statistical significance of the models in terms of accuracy and string edit distance, respectively. To test the statistical significance of the BLEU scores of the models, we used a bootstrap resampling together with an approximate randomization method (Clark et al., 2011)[†].

6.6.3 Settings

NeuralREG was implemented using Dynet (Neubig et al., 2017). Source and target word embeddings were 300D each and trained jointly with the model, whereas hidden units are 512D for each direction, totaling 1024D in the bidirection layers. All non-recurrent matrices were initialized following the method of Glorot & Bengio (2010).

Models were trained using stochastic gradient descent with Adadelta (Zeiler, 2012) and mini-batches of size 40. We ran each model for 60 epochs, applying early stopping for model selection based on accuracy on the development set with patience of 20 epochs. For each decoding version (`Seq2Seq`, `CAtt` and `HierAtt`), we search for the best combination of drop-out probability of 0.2 or 0.3 in both the encoding and decoding layers,

[†] `https://github.com/jhclark/multeval`

using beam search with a size of 1 or 5 with predictions up to 30 tokens or until 2 ending tokens were predicted (*EOS*). The results described in the next section were obtained on the test set by the NeuralREG version with the highest accuracy on the development set over the epochs.

6.7 Results

Tables 6.1 summarize the results for all models on all metrics on the test set and Table 6.2 depicts a text example lexicalized by each model. The first thing to note in the results of the first table is that the baselines in the top two rows perform quite strong on this task, generating more than half of the referring expressions exactly as in the gold-standard. The method based on Castro Ferreira et al. (2016b) (Chapter 3) performs better on all metrics, and is also capable, albeit to a limited extent, to predict pronominal references (which the *OnlyNames* baseline obviously cannot).

We reported results on the test set for NeuralREG+`Seq2Seq` and NeuralREG+`CAtt` using dropout probability 0.3 and beam size 5, and NeuralREG+`HierAtt` with dropout probability of 0.3 and beam size of 1 selected based on the highest accuracy on the development set. Importantly, the three NeuralREG variant models statistically outperformed the two baseline systems. They achieved BLEU scores, text and referential accuracies as well as string edit distances in the range of 79.01-79.39, 28%-30%, 73%-74% and 2.25-2.36, respectively. This means that NeuralREG predicted 3 out of 4 references completely correct, whereas the incorrect ones needed an average of 2 post-edition operations in character level to be equal to the gold-standard. When considering the texts lexicalized with the referring expressions produced by NeuralREG, at least 28% of them are similar to the original texts. Especially noteworthy was the score on

	All References	
	Accuracy	SED
OnlyNames	0.53^{D}	4.05^{D}
Ferreira	0.61^{C}	3.18^{C}
NeuralREG+`Seq2Seq`	$0.74^{A,B}$	$2.32^{A,B}$
NeuralREG+`CAtt`	0.74^{A}	2.25^{A}
NeuralREG+`HierAtt`	0.73^{B}	2.36^{B}

	Pronouns			
	Acc.	Prec.	Rec.	F-Score
OnlyNames	-	-	-	-
Ferreira	0.43^{B}	0.57	0.54	0.55
NeuralREG+`Seq2Seq`	0.75^{A}	0.77	0.78	0.78
NeuralREG+`CAtt`	0.75^{A}	0.73	0.78	0.75
NeuralREG+`HierAtt`	0.73^{A}	0.74	0.77	0.75

	Text	
	Accuracy	BLEU
OnlyNames	0.15^{D}	69.03^{D}
Ferreira	0.19^{C}	72.78^{C}
NeuralREG+`Seq2Seq`	0.28^{B}	$79.27^{A,B}$
NeuralREG+`CAtt`	0.30^{A}	79.39^{A}
NeuralREG+`HierAtt`	$0.28^{A,B}$	79.01^{B}

Table 6.1: (1) Accuracy and String Edit Distance (SED) results in the prediction of all referring expressions; (2) Accuracy (Acc.), Precision (Prec.), Recall (Rec.) and F-Score results in the prediction of pronominal forms; and (3) Accuracy and BLEU score results of the texts with the generated referring expressions. Rankings were determined by statistical significance.

Model	Text
Original	**alan shepard** was born in **new hampshire** on **18 november 1923** . before **his** death in **california he** had been awarded **the distinguished service medal by the us navy** an award higher than **the department of commerce gold medal** .
`OnlyNames`	**alan shepard** was born in **new hampshire** on **1923-11-18** . before **alan shepard** death in **california alan shepard** had been awarded **distinguished service medal (united states navy)** an award higher than **department of commerce gold medal** .
Ferreira	**alan shepard** was born in **new hampshire** on **1923-11-18** . before **alan shepard** death in **california it** had been awarded **distinguished service medal** an award higher than **department of commerce gold medal** .
`Seq2Seq`	**alan shepard** was born in **new hampshire** on **1923-11-18** . before **his** death in **california him** had been awarded **the distinguished service medal by the united states navy** an award higher than **the department of commerce gold medal** .
`CAtt`	**alan shepard** was born in **new hampshire** on **1923-11-18** . before **his** death in **california he** had been awarded **the distinguished service medal by the us navy** an award higher than **the department of commerce gold medal** .
`HierAtt`	**alan shephard** was born in **new hampshire** on **1923-11-18** . before **his** death in **california he** had been awarded **the distinguished service medal** an award higher than **the department of commerce gold medal** .

Table 6.2: Example of text with references produced by each model.

pronoun accuracy, indicating that the model was well capable of predicting when to generate a pronominal reference in our dataset.

The results for the different decoding methods for NeuralREG were similar, with the NeuralREG+`CAtt` performing slightly better in terms of the BLEU score, text accuracy and String Edit Distance. The more complex NeuralREG+`HierAtt` yielded the lowest results, even though the differences with the other two models were small and not even statistically significant in many of the cases.

6.8 Human evaluation

As a complement to the automatic evaluation, we performed an evaluation with human judges, comparing the quality judgments of the original texts to the versions generated by our various models.

6.8.1 Material

We quasi-randomly selected 24 instances from the delexicalized version of the WebNLG corpus related to the test part of the referring expression collection. For each of the selected instances, we took into account its source triple set and its 6 target texts: one original (randomly chosen) and its versions with the referring expressions generated by each of the 5 models introduced in this chapter (two baselines, three neural models). Instances were chosen following 2 criteria: the number of triples in the source set (ranging from 2 to 7) and the differences between the target texts.

For each size group, we randomly selected 4 instances (of varying degrees of variation between the generated texts) giving rise to 144 trials ($=$ 6 triple set sizes $*$ 4 instances $*$ 6 text versions), each consisting of a

set of triples and a target text describing it with the lexicalized referring expressions highlighted in yellow.

6.8.2 Method

The experiment had a latin-square design, distributing the 144 trials over 6 different lists such that each participant rated 24 trials, one for each of the 24 corpus instances, making sure that participants saw equal numbers of triple set sizes and generated versions. Once introduced to a trial, the participants were asked to rate the fluency ("does the text flow in a natural, easy to read manner?"), grammaticality ("is the text grammatical (no spelling or grammatical errors)?") and clarity ("does the text clearly express the data?") of each target text on a 7-Likert scale, focusing on the highlighted referring expressions. The experiment is available in the website of the author[‡].

6.8.3 Participants

We recruited 60 participants, 10 per list, via Mechanical Turk. Their average age was 36 years and 27 of them were females. The majority declared themselves native speakers of English (44), while 14 and 2 self-reported as fluent or having a basic proficiency, respectively.

6.8.4 Results

Table 6.3 summarizes the result. Inspection of the Table reveals a clear pattern: all three neural models scored higher than the baselines on all metrics, with especially NeuralREG+`CAtt` approaching the ratings for the

[‡]`https://ilk.uvt.nl/~tcastrof/acl2018/evaluation/`

	Fluency	Grammar	Clarity
OnlyNames	4.74^{C}	4.68^{B}	4.90^{B}
Ferreira	4.74^{C}	4.58^{B}	4.93^{B}
NeuralREG+`Seq2Seq`	$4.95^{B,C}$	$4.82^{A,B}$	4.97^{B}
NeuralREG+`CAtt`	$5.23^{A,B}$	$4.95^{A,B}$	$5.26^{A,B}$
NeuralREG+`HierAtt`	$5.07^{B,C}$	$4.90^{A,B}$	$5.13^{A,B}$
Original	5.41^{A}	5.17^{A}	5.42^{A}

Table 6.3: Fluency, Grammaticality and Clarity results obtained in the human evaluation. Rankings were determined by statistical significance.

original sentences, although – again – differences between the neural models were small. Concerning the size of the triple sets, we did not find any clear pattern.

To test the statistical significance of the pairwise comparisons, we used the Wilcoxon signed-rank test corrected for multiple comparisons by the Bonferroni method. Different from the automatic evaluation, the results of both baselines were not statistically significant for the three metrics. In comparison with the neural models, NeuralREG+`CAtt` statistically outperformed the baselines in terms of fluency, whereas the other comparisons among baselines and neural models were not statistically significant. The results for the 3 different decoding methods of NeuralREG also did not reveal a significant difference. Finally, the original texts were rated significantly higher than both baselines in terms of the three metrics, also than NeuralREG+`Seq2Seq` and NeuralREG+`HierAtt` in terms of fluency, and than NeuralREG+`Seq2Seq` in terms of clarity.

	Fluency	Grammar	Clarity
BLEU	0.14	0.14	0.02
Fluency	-	0.85*	0.91*
Grammar	-	-	0.80*

Table 6.4: Spearman's rank correlation coefficients between the BLEU score and the human evaluation measures Fluency, Grammaticality and Clarity. Statistically significant results at $p < 0.01$ are denoted by "*".

6.9 Relation between evaluations

To some extent, our automatic and human evaluations seem to point in different directions. To get a better appreciation of these differences, we computed the Spearman's rank correlations between the automatic metric BLEU and the 3 ratings given by the participants in the human evaluation (fluency, grammaticality and clarity).

6.9.1 Method

We used the same 144 trials of the human evaluation, where 24 trials were original texts, and 120 trials were the alternative versions with referring expressions produced by each of our 5 models. A BLEU score was obtained for each text version in comparison with its original counterpart. To obtain the fluency, grammaticality and clarity of the same texts, we used the average ratings of the participants for each one of them. Finally, Spearman's rank correlation coefficients were computed over the matrix of 120 (trials) by 4 (measures).

6.9.2 Result

Table 6.4 shows the Spearman's rank correlation coefficients for each pair of metrics. The automatic metric BLEU shows a slight positive correlation with the measures of the human evaluation. However, none of these were statistically significant. On the other hand, when looking at the human evaluation measures, it can be seen that all of these correlated strongly (and significantly).

6.10 Discussion

This chapter introduced NeuralREG, an end-to-end approach based on neural networks which tackles the full Referring Expression Generation process. It generates referring expressions for discourse entities by simultaneously selecting form and content without any need of feature extraction techniques. The model was implemented using an encoder-decoder approach where a target referent and its surrounding linguistic contexts were first encoded and combined into a single vector representation which subsequently was decoded into a referring expression to the target, suitable for the specific discourse context. In an automatic evaluation on a collection of 78,901 referring expressions to 1,501 Wikipedia entities, the different versions of the model all yielded better results than the two (competitive) baselines. Later in a complementary human evaluation, the texts with referring expressions generated by a variant of our novel model were considered statistically more fluent than the texts lexicalized by the two baselines.

Data The collection of referring expressions used in our experiments was extracted from a novel, delexicalized and publicly available version of

the WebNLG corpus (Gardent et al., 2017a,b), where the discourse entities were replaced with general tags for decreasing the data sparsity. Besides the REG task, these data can be useful for many other tasks related to, for instance, the NLG process (Reiter & Dale, 2000; Gatt & Krahmer, 2018) and Wikification (Moussallem et al., 2017).

Baselines We introduced two strong baselines which generated roughly half of the referring expressions identical to the gold standard in an automatic evaluation. These baselines performed relatively well because they frequently generated full names, which occur often for our wikified references. However, they performed poorly when it came to pronominalization, which is an important ingredient for fluent, coherent text. *OnlyNames*, as the name already reveals, does not manage to generate any pronouns. However, the approach of Castro Ferreira et al. (2016b) (Chapter 3) also did not perform well in the generation of pronouns, revealing a poor capacity to detect highly salient entities in a text.

NeuralREG was implemented with 3 different decoding architectures: `Seq2Seq`, `CAtt` and `HierAtt`. Although all the versions performed relatively similar, the concatenative-attention (`CAtt`) version generated the closest referring expressions from the gold-standard ones and presented the highest textual accuracy in the automatic evaluation. The texts lexicalized by this variant were also considered statistically more fluent than the ones generated by the two proposed baselines in the human evaluation.

Surprisingly, the most complex variant (`HierAtt`) with a hierarchical-attention mechanism gave lower results than `CAtt`, producing lexicalized texts which were rated as less fluent than the original ones and not significantly more fluent from the ones generated by the baselines. This result appears to be not consistent with the findings of Libovický & Helcl

(2017), who reported better results on multi-modal machine translation with hierarchical-attention as opposed to the flat variants (Specia et al., 2016).

Finally, our NeuralREG variant with the lowest results were our 'vanilla' sequence-to-sequence (`Seq2Seq`), whose the lexicalized texts were significantly less fluent and clear than the original ones. This shows the importance of the attention mechanism in the decoding step of NeuralREG in order to generate fine-grained referring expressions in discourse.

Automatic vs. Human evaluations In an error analysis, we measured the correlation among the automatic metric BLEU and the ratings in the human evaluation. The results are in agreement with other studies in the literature (Stent et al., 2005; Belz & Reiter, 2006; Novikova et al., 2017a), which showed a weak or no correlation among automatic metrics like BLEU and human judgments, highlighting the importance of the human evaluation and the need for better automatic metrics.

Conclusion We introduced a deep learning model for the generation of referring expressions in discourse texts. NeuralREG decides both on referential form and on referential content in an integrated, end-to-end approach, without using explicit features. Using a new delexicalized version of the WebNLG corpus (made publicly available), we showed that the neural model substantially improves over two strong baselines in terms of accuracy of the referring expressions and fluency of the lexicalized texts.

7

Linguistic realization as machine translation: Comparing different MT models for AMR-to-text generation

Abstract In this chapter, we study AMR-to-text generation, framing it as a translation task and comparing two different MT approaches (Phrase-based and Neural MT). We systematically study the effects of 3 AMR pre-processing steps (Delexicalization, Compression, and Linearization) applied before the MT phase. Our results show that preprocessing indeed helps, although the benefits differ for the two MT models. Data and models are publicly available*.

*https://github.com/ThiagoCF05/LinearAMR

This chapter is based on Castro Ferreira, T., Calixto, I., Wubben, S., & Krahmer, E. (2017). Linguistic realization as machine translation: Comparing different MT models for AMR-to-text generation. In *Proceedings of the 10th International Conference on Natural Language Generation*, INLG'2017 (pp. 1–10). Santiago de Compostela, Spain: Association for Computational Linguistics.

7.1 Introduction

Natural Language Generation (NLG) is the process of generating coherent natural language text from non-linguistic data (Reiter & Dale, 2000). While there is broad consensus among NLG scholars on the output of NLG systems (i.e., text), there is far less agreement on what the input should be; see Gatt & Krahmer (2018) for a recent review. Over the years, NLG systems have taken a wide range of inputs, including for example images (Xu et al., 2015), numeric data (Gkatzia et al., 2014) and semantic representations (Theune et al., 2001).

This chapter focuses on generating natural language based on Abstract Meaning Representations (AMRs) (Banarescu et al., 2013). AMRs encode the meaning of a sentence as a rooted, directed and acyclic graph, where nodes represent concepts, and labeled directed edges represent relations among these concepts. The formalism strongly relies on the PropBank notation. Figure 7.1 shows an example.

AMRs have increased in popularity in recent years, partly because they are relatively easy to produce, to read and to process automatically. In addition, they can be systematically translated into first-order logic, allowing for a well-specified model-theoretic interpretation (Bos, 2016). Most earlier studies on AMRs have focused on text understanding, i.e. processing texts in order to produce AMRs (Flanigan et al., 2014; Artzi et al., 2015). However, recently the reverse process, i.e. the generation of texts from AMRs, has started to receive scholarly attention (Flanigan et al., 2016; Song et al., 2016; Pourdamghani et al., 2016; Song et al., 2017; Konstas et al., 2017).

We assume that in practical applications, conceptualization models or dialogue managers (models which decide "*what to say*") output AMRs. In this paper we study different ways in which these AMRs can be converted

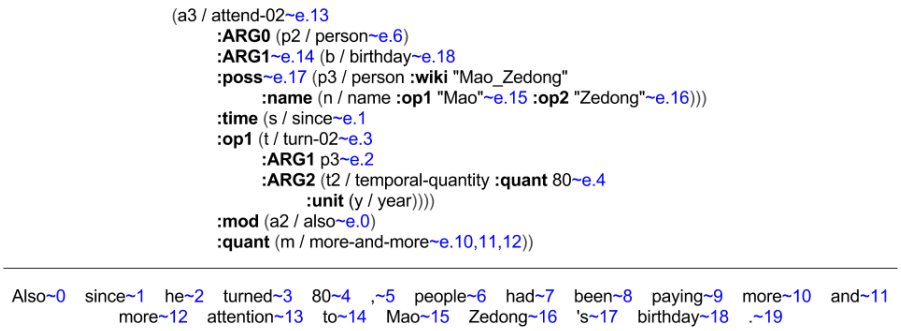


Figure 7.1: Example of an AMR

into natural language (deciding "*how to say it*"). We approach this as a translation problem—automatically translating from AMRs into natural language—and the key-contribution of this paper is that we systematically compare different preprocessing strategies for two different MT systems: Phrase-based MT (PBMT) and Neural MT (NMT).

We look at potential benefits of three preprocessing steps on AMRs before feeding them into an MT system: *delexicalization*, *compression*, and *linearization*. Delexicalization decreases the sparsity of an AMR by removing constant values, compression removes nodes and edges which are less likely to be aligned to any word on the textual side and linearization 'flattens' the AMR in a specific order. Combining all possibilities gives rise to $2^3 = 8$ AMR preprocessing strategies, which we evaluate for two different MT systems: PBMT and NMT.

Following earlier work in AMR-to-text generation and the MT literature, we evaluate the system outputs in terms of fluency, adequacy and post-editing effort, using BLEU (Papineni et al., 2002), METEOR (Lavie & Agarwal, 2007) and TER (Snover et al., 2006) scores, respectively. We show that preprocessing helps, although the extent of the benefits differs for the two MT systems.

7.2 Related work

To the best of our knowledge, Flanigan et al. (2016) was the first study that introduced a model for natural language generation from AMRs. The model consists of two steps. First, the AMR-graph is converted into a spanning tree, and then, in a second step, this tree is converted into a sentence using a tree transducer.

In Song et al. (2016), the generation of a sentence from an AMR is addressed as an asymmetric generalized traveling salesman problem (AGTSP). For sentences shorter than 30 words, the model does not beat the system described by Flanigan et al. (2016). However, Song et al. (2017) treat the AMR-to-text task using a Synchronous Node Replacement Grammar (SNRG) and outperform Flanigan et al. (2016).

Although AMRs do not contain articles and do not represent inflectional morphology for tense and number (Banarescu et al., 2013), the formalism is relatively close to the (English) language. Motivated by this similarity, Pourdamghani et al. (2016) proposed an AMR-to-text method that organizes some of these concepts and edges in a flat representation, commonly known as *Linearization*. Once the linearization is complete, Pourdamghani et al. (2016) map the flat AMR into an English sentence using a Phrase-Based Machine Translation (PBMT) system. This method yields better results than Flanigan et al. (2016) on development and test set from the LDC2014T12 corpus.

Pourdamghani et al. (2016) train their system using a set of AMR-sentence pairs obtained by the aligner described in Pourdamghani et al. (2014). In order to decrease the sparsity of the AMR formalism caused by the ratio of broad vocabulary and relatively small amount of data, this aligner drops a considerable amount of the AMR structure, such as role edges :ARG0, :ARG1, :mod, etc. However, inspection of the gold-standard

alignments provided in the LDC2016E25 corpus revealed that this rule-based compression can be harmful for the generation of sentences, since such role edges can actually be aligned to function words in English sentences. So having these roles available arguably could improve AMR-to-text translation. This indicates that a better comparison of the effects of different preprocessing steps is called for, which we do in this chapter.

In addition, Pourdamghani et al. (2016) use PBMT, which is devised for translation but also utilized in other NLP tasks, e.g. text simplification (Wubben et al., 2012; Štajner et al., 2015). However, these systems have the disadvantage of having many different feature functions, and finding optimal settings for all of them increases the complexity of the problem from an engineering point of view.

An alternative MT model has been proposed: Neural Machine Translation (NMT). NMT models frame translation as a sequence-to-sequence problem (Bahdanau et al., 2015), and have shown strong results when translating between many different language pairs (Bojar et al., 2015). Recently, Konstas et al. (2017) introduce sequence-to-sequence models for parsing (text-to-AMR) and generation (AMR-to-text). They use a semi-supervised training procedure, incorporating 20M English sentences which do not have a gold-standard AMR, thus overcoming the limited amount of data available. They report state-of-the-art results for the task, which suggests that NMT is a promising alternative for AMR-to-text.

7.3 Models

We describe our AMR-to-text generation models, which rely on 3 preprocessing steps (*delexicalization*, *compression*, and/or *linearization*) followed by a machine translation and realization steps.

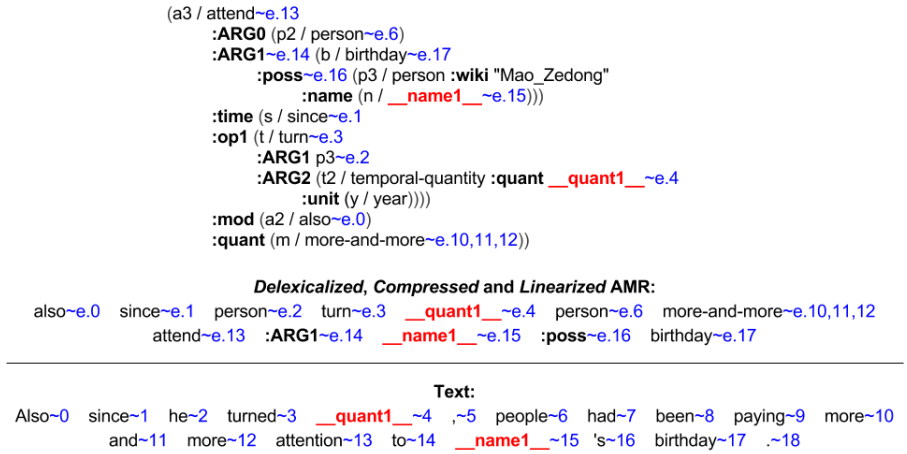


Figure 7.2: Example of a *Delexicalized*, *Compressed* and *Linearized* AMR

7.3.1 Delexicalization

Inspection of the LDC2016E25 corpus reveals that on average 22% of the structure of an AMR are AMR constant values, such as names, quantities, and dates. This information increases the sparsity of the data, and makes it arguably more difficult to map an AMR into a textual format. To address this, Pourdamghani et al. (2016) look for special realization component for names, dates and numbers in development and test sets and add them on the training set. On the other hand, similar to Konstas et al. (2017), we delexicalized these constants, replacing the original information for tags (e.g., `__name1__`, `__quant1__`). A list of tag-values is kept, aiming to identifying the position and to insert the original information in the sentence after the translation step is completed. Figure 7.2 shows a delexicalized AMR.

7.3.2 Compression

Given the alignment between an AMR and a sentence, the nodes and edges in the AMR can either be aligned to words in the sentence or not. So before the linearization step, we would like to know which elements of an AMR should actually be part of the 'flattened' representation.

Following the aligner of Pourdamghani et al. (2014), Pourdamghani et al. (2016) clean an AMR by removing some nodes and edges independent of the context. Instead, we are using alignments that may relate a given node or edge to an English word according to the context. In Figure 7.1 for instance, the first edge :ARG1 is aligned to the preposition *to* from the sentence, whereas the second edge with a similar value is not aligned to any word in the sentence. Therefore, we need to train a classifier to decide which parts of an AMR should be in the flattened representation according to the context.

To solve the problem, we train a Conditional Random Field (CRF) which determines whether a node or an edge of an AMR should be included in the flattened representation. The classification process is sequential over a flattened representation of an AMR obtained by depth first search through the graph. Each element is represented by their name and parent name. We use *CRFSuite* (Okazaki, 2007) to implement our model.

7.3.3 Linearization

After Compression, we flatten the AMR to serve as input to the translation step, similarly as proposed in Pourdamghani et al. (2016). We perform a depth-first search through the AMR, printing the elements according to their visiting order. In a second step, also following Pourdamghani et al. (2016), we implemented a version of the 2-Step Classifier from Lerner

& Petrov (2013) to preorder the elements from an AMR according to the target side.

2-Step Classifier We implement the preordering method proposed by Lerner & Petrov (2013) in the following way. We define the order among a head node and its subtrees in two steps. In the first, we use a trained maximum entropy classifier to predict for each subtree whether it should occur before or after the head node. As features, we represent the head node by its frameset, whereas the subtree is represented by its head node frameset and parent edge.

Once we divide the subtrees into the ones which should occur before and after the head node, we use a maximum entropy classifier for the size of the subtree group to predict their order. For instance, for a group of 2 subtrees, a maximum entropy classifier specific for groups of 2 subtrees would be used to predict the permutation order of them (0-1 or 1-0). As features, the head node is also represented by its PropBank frameset, whereas the subtrees of the groups are represented by their parent edges, their head node framesets and by which side of the head node they are (before or after). We train classifiers for groups of sizes between 2 and 4 subtrees. For bigger groups, we used the depth first search order.

7.3.4 Translation models

To map a flat AMR representation into an English sentence, we use phrase-based (Koehn et al., 2003) and neural machine translation (Bahdanau et al., 2015) models.

Phrase-based machine translation

These models use Bayes rule to formalize the problem of translating a text from a source language f to a target language e. In our case, we want to translate a flat amr into an English sentence e as Equation 7.1 shows.

$$P(e \mid amr) = argmax\ P(amr \mid e)P(e) \tag{7.1}$$

The *a priori* function $P(e)$ usually is represented by a language model trained on the target language. The *a posteriori* equation is calculated by the log-linear model described at Equation 7.2.

$$P(amr \mid e) = argmax \sum_{j=1}^{J} \lambda_j h_j(amr, e) \tag{7.2}$$

Each $h_j(amr, e)$ is an arbitrary feature function over AMR-sentence pairs. To calculate it, the flat amr is segmented into I phrases $a\bar{m}r_1^I$, such that each phrase $a\bar{m}r_i$ is translated into a target phrase $\bar{e}_i$ as described by Equation 7.3.

$$h_j(amr, e) = argmax\ h_j(a\bar{m}r_i^I, \bar{e}_i^I) \tag{7.3}$$

As feature functions, we used direct and inverse phrase translation probabilities and lexical weighting; word, unknown word and phrase penalties.

We also used models to reorder a flat amr according to the target sentence e at decoding time. They work on the word-level (Koehn et al., 2003), at the level of adjacent phrases (Koehn et al., 2005) and beyond adjacent phrases (hierarchical-level) (Galley & Manning, 2008). Phrase- and hierarchical level models are also known as lexicalized reordering models.

As Koehn et al. (2003), given s_i the start position of the source phrase $a\bar{m}r_i$ translated into the English phrase $\bar{e}_i$, and f_{i-1} the end position of the source phrase $a\bar{m}r_{i-1}$ translated into the English phrase $\bar{e}_{i-1}$, a distortion model $\alpha^{|s_i - f_{i-1} - 1|}$ is defined as a distance-based reordering model. α is chosen by tunning the model.

Lexicalized reordering models are more complex than distance-based ones, but usually help the system to obtain better results (Koehn et al., 2005; Galley & Manning, 2008). Given a possible set of target phrases $e = (\bar{e}_1, \ ... \ , \bar{e}_n)$ based on a source amr, and a set of alignments $a = (a_1, \ ... \ , a_n)$ that maps a source phrase $a\bar{m}r_{a_i}$ into a target phrase $\bar{e}_i$, a lexicalized model aims to predict a set of orientations $o = (o_1, \ ... \ , o_n)$ as Equation 7.4 shows.

$$P(o \mid e, amr) = \prod_{i=1}^{n} P(o_i \mid \bar{e}_i, a\bar{m}r_{a_i}, a_{i-1}, a_i) \tag{7.4}$$

Each orientation o_i, attached to the hypothesized target phrase e_i, can be a monotone (M), swap (S) or discontinuous (D) operation according to Equation 7.5.

$$o_i = \begin{cases} M, & \text{if } a_i - a_{i-1} = 1 \\ S, & \text{if } a_i - a_{i-1} = -1 \\ D, & \text{if } |a_i - a_{i-1}| \neq 1 \end{cases} \tag{7.5}$$

In the hierarchical model, we distinguished the discontinuous operation by the direction: discontinuous right ($a_i - a_{i-1} < 1$) and discontinuous left ($a_i - a_{i-1} > 1$). These models are important for our task, since the preordering method used in the Linearization step can be insufficient to adequate it to the target sentence order.

Neural machine translation

Following the attention-based Neural Machine Translation (NMT) model introduced by Bahdanau et al. (2015), given a flat $amr = (amr_1, amr_2, \cdots, amr_N)$ and its English sentence translation $e = (e_1, e_2, \cdots, e_M)$, a single neural network is trained to translate amr into e by directly learning to model $p(e \mid amr)$. The network consists of one *encoder*, one *decoder*, and one *attention mechanism*.

The encoder is a bi-directional RNN with gated recurrent units (GRU) (Cho et al., 2014), where one forward RNN $\overrightarrow{\Phi}_{\text{enc}}$ reads the amr from left to right and generates a sequence of *forward annotation vectors* $(\overrightarrow{\boldsymbol{h}}_1, \overrightarrow{\boldsymbol{h}}_2, \cdots, \overrightarrow{\boldsymbol{h}}_N)$ at each encoder time step $i \in [1, N]$, and a backward RNN $\overleftarrow{\Phi}_{\text{enc}}$ reads the amr from right to left and generates a sequence of *backward annotation vectors* $(\overleftarrow{\boldsymbol{h}}_N, \overleftarrow{\boldsymbol{h}}_{N-1}, \cdots, \overleftarrow{\boldsymbol{h}}_1)$. The final annotation vector is the concatenation of forward and backward vectors $\boldsymbol{h}_i = \left[\overrightarrow{\boldsymbol{h}_i}; \overleftarrow{\boldsymbol{h}_i}\right]$, and $C = (\boldsymbol{h}_1, \boldsymbol{h}_2, \cdots, \boldsymbol{h}_N)$ is the set of source annotation vectors.

The decoder is a neural LM conditioned on the previously emitted words and the source sentence via an attention mechanism over C. A multilayer perceptron is used to initialize the decoder's hidden state $\boldsymbol{s}_0$, where the input to this network is the concatenation of the last forward and backward vectors $\left[\overrightarrow{\boldsymbol{h}_N}; \overleftarrow{\boldsymbol{h}_1}\right]$.

At each time step t of the decoder, we compute a *time-dependent* context vector $\boldsymbol{c}_t$ based on the annotation vectors C, the decoder's previous hidden state $\boldsymbol{s}_{t-1}$ and the target English word $\tilde{e}_{t-1}$ emitted by the decoder in the previous time step. A single-layer feed-forward network computes an *expected alignment* $a_{t,i}$ between each source annotation vector $\boldsymbol{h}_i$ and

the target word to be emitted at the current time step t, as in (7.6):

$$a_{t,i} = \boldsymbol{v}_a{}^T \tanh(\boldsymbol{U}_a \boldsymbol{s}_{t-1} + \boldsymbol{W}_a \boldsymbol{h}_i). \tag{7.6}$$

In Equation (7.7), these expected alignments are normalized and converted into probabilities:

$$\alpha_{t,i} = \frac{\exp(a_{t,i})}{\sum_{j=1}^{N} \exp(a_{t,j})}, \tag{7.7}$$

where $\alpha_{t,i}$ are called the model's *attention weights*, which are in turn used in computing the time-dependent context vector $\boldsymbol{c}_t = \sum_{i=1}^{N} \alpha_{t,i} \boldsymbol{h}_i$. Finally, the context vector $\boldsymbol{c}_t$ is used in computing the decoder's hidden state $\boldsymbol{s}_t$ for the current time step t, as shown in Equation (7.8):

$$\boldsymbol{s}_t = \Phi_{\text{dec}}(\boldsymbol{s}_{t-1}, \boldsymbol{W}_{\boldsymbol{e}}[\tilde{e}_{t-1}], \boldsymbol{c}_t), \tag{7.8}$$

where $\boldsymbol{s}_{t-1}$ is the decoder's previous hidden state, $\boldsymbol{W}_{\boldsymbol{e}}[\tilde{e}_{t-1}]$ is the embedding of the word emitted in the previous time step, and $\boldsymbol{c}_t$ is the updated time-dependent context vector. Given a hidden state $\boldsymbol{s}_t$, the probabilities for the next target word are computed using one projection layer followed by a softmax, as illustrated in 7.9, where the matrices $\boldsymbol{L}_o$, $\boldsymbol{L}_s$, $\boldsymbol{L}_w$ and $\boldsymbol{L}_c$ are transformation matrices and $\boldsymbol{c}_t$ is the time-dependent context vector.

$$p(e_t = k \mid \boldsymbol{e}_{<\boldsymbol{t}}, \boldsymbol{c}_{\boldsymbol{t}}) \propto \exp(\boldsymbol{L}_{\boldsymbol{o}} \tanh(\boldsymbol{L}_{\boldsymbol{s}} \boldsymbol{s}_t + \boldsymbol{L}_{\boldsymbol{w}} \boldsymbol{E}_{\boldsymbol{e}}[\hat{e}_{t-1}] + \boldsymbol{L}_{\boldsymbol{c}} \boldsymbol{c}_{\boldsymbol{t}})). \tag{7.9}$$

7.3.5 Realization

Since we delexicalize names, dates, quantities and values from AMRs, we need to textually realize this information once we obtain the results from the translation step. As we kept all the original information and their relation with the tags, we just need to replace one for the other.

We implement some rules to adequate our generated texts to the ones we saw in the training set. Different from the AMRs, we represent months nominally, and not numerically - month *5* will be *May* for example. Values and quantities bigger than a thousand are also part realized nominally. The value *8500000000* would be realized as *8.5 billion* for instance. On the other hand, names are realized as they are.

7.4 Evaluation

7.4.1 Data

We used the corpus LDC2016E25 provided by the SemEval 2017 Task 9 in our evaluation. This corpus consists of aligned AMR-sentence pairs, mostly newswire. We considered the train/dev/test sets splitting proposed in the original setting, totaling 36,521, 1,368 and 1,371 AMR-sentence pairs, respectively. Compression and Linearization methods, as well as Phrase-based Machine Translation models were trained over the gold-standard alignments between AMRs and sentences on the training set of the corpus.

7.4.2 Evaluated models

We test models with and without the Delexicalization/Realization (-Delex and +Delex) and Compression (-Compress and +Compress) steps. In mod-

els without the Compression step, we include all the elements from an AMR in the flattened representation. For the Linearization step, we flatten the AMR structure based on a depth-first search (-Preorder) or preordering it with our 2-step classifier (+Preorder). Finally, we translate a flattened AMR into text using a Phrase-based (PBMT) and a Neural Machine Translation model (NMT). In total, we evaluated 16 models.

Phrase-based Machine Translation We used a standard PBMT system built using Moses toolkit (Koehn et al., 2007). At training time, we extract and score phrase sentences up to the size of 9 tokens. All the feature functions were trained using the gold-standard alignments from the training set and their weights were tuned on the development data using k-batch MIRA with $k = 60$ (Cherry & Foster, 2012) with BLEU as the evaluation metric. A distortion limit of 6 was used for the reordering models. Lexicalized reordering models were bidirectional. At decoding time, we use a stack size of 1000.

Our language model $P(e)$ is a 5-gram LM trained on the Gigaword Third Edition corpus using KenLM (Heafield et al., 2013). For the models with the Delexicalization step, we trained the language model with a delexicalized version of Gigaword by parsing the corpus using the Stanford Named Entity Recognition tool (Finkel et al., 2005). All the entities labeled as `LOCATION`, `PERSON`, `ORGANISATION` or `MISC` were replaced by the tag `__nameX__`. Entities labeled as `NUMBER` or `MONEY` were replaced by the tag `__quantX__`. Finally, entities labeled as `PERCENT` or `ORDINAL` were replaced by `__valueX__`. In the tags, `X` is replaced by the ordinal position of the entity in the sentence.

Neural Machine Translation The encoder is a bidirectional RNN with GRU, each with a 1024D hidden unit. Source and target word embed-

dings are 620D each and are both trained jointly with the model. All non-recurrent matrices are initialized by sampling from a Gaussian ($\mu = 0, \sigma = 0.01$), recurrent matrices are random orthogonal and bias vectors are all initialized to zero. The decoder RNN also uses GRU and is a neural LM conditioned on its previous emissions and the source sentence by means of the source attention mechanism.

We apply dropout with a probability of 0.3 in both source and target word embeddings, in the encoder and decoder RNNs inputs and recurrent connections, and before the readout operation in the decoder RNN. We follow Gal & Ghahramani (2016) and apply dropout to the encoder and decoder RNNs using the same mask in all time steps.

Models are trained using stochastic gradient descent with Adadelta (Zeiler, 2012) and minibatches of size 40. We apply early stopping for model selection based on BLEU scores, so that if a model does not improve on the validation set for more than 20 epochs, training is halted.

7.4.3 Models for comparison

We compare BLEU scores for some of the AMR-to-text systems described in the literature (Flanigan et al., 2016; Song et al., 2016; Pourdamghani et al., 2016; Song et al., 2017; Konstas et al., 2017). Since the models of Flanigan et al. (2016) and Pourdamghani et al. (2016) are publicly available, we also use them with the same training data as our models. For Flanigan et al. (2016), we specifically use the version available on GitHub[†].

For Pourdamghani et al. (2016), we use the version available at the first author's website[‡]. The rules used for the preordering model and the fea-

[†] http://github.com/jflanigan/jamr/tree/Generator
[‡] http://isi.edu/~damghani/papers/amr2eng.zip

ture functions from the PBMT system are trained using alignments over AMR–sentence pairs from the training set obtained with the aligner described by Pourdamghani et al. (2014). We do not use lexicalized reordering models as Pourdamghani et al. (2016). Moreover, we tune the weights of the feature functions with MERT (Och, 2003).

Both models make use of a 5-gram language model trained on Gigaword Third Edition corpus with KenLM.

7.4.4 Metrics

To evaluate fluency, adequacy and post-editing effort of the models, we use BLEU (Papineni et al., 2002), METEOR (Lavie & Agarwal, 2007) and TER (Snover et al., 2006), respectively.

7.5 Results

Table 7.1 depicts the scores of the different models by the size of the data they were trained on. For illustration, we depicted the BLEU scores of all the AMR-to-text systems described in the literature. The models of Flanigan et al. (2016) and Pourdamghani et al. (2016) were officially trained with 10,313 AMR-sentence pairs from the LDC2014T12 corpus, and with 36,521 AMR-sentence pairs from the LDC2016E25 in our study (as our models). The ones of Song et al. (2016) and Song et al. (2017) were trained with 16,833 pairs from the LDC2015E86 corpus. Konstas et al. (2017), which presents the highest quantitative result in the task so far, also used the LDC2015E86 corpus plus 20 million English sentences from the Gigaword corpus with a semi-supervised approach. We report the results when their model were trained only with AMR-sentence pairs from the corpus, and when improved with more 20 million sentences.

Among the PBMT models, the Delexicalization step (+Delex) does not seem to play a role in obtaining better sentences from AMRs. All the models with the preordering method in Linearization (+Preorder) introduce better results than Flanigan et al. (2016) and Song et al. (2016), whereas only the lexicalized models with the preordering method (PBMT+Delex[+|-]Compress+Preorder) outperform Song et al. (2017) and introduce competitive results with Pourdamghani et al. (2016).

In our NMT models, apparently the Compression step is harmful to the task, whereas Delexicalization and preordering in Linearization lead to better results. However, none of the NMT models outperform neither the PBMT models nor the baselines.

7.6 Discussion

In this paper, we studied models for AMR-to-text generation using machine translation. We systematically analyzed the effects of 3 processing strategies on AMRs before feeding them either to a Phrase-based or a Neural MT system. The evaluation was performed on the LDC2016E25 corpus, provided by SemEval 2017 Task 9. All the models had the fluency, adequacy and post-editing effort of their produced sentences measured by BLEU, METEOR and TER, respectively. In general, we found that processing AMRs helps, although the effects differ for the different systems.

Phrase-based MT Delexicalization (+Delex) does not seem to play a role in obtaining better sentences from AMRs using PBMT. Our best model (PBMT-Delex+Compress+Preorder) presents competitive results to Pourdamghani et al. (2016) with the advantage that no technique is necessary to overcome data sparsity.

	Data Size	BLEU	METEOR	TER
(Flanigan et al., 2016)	∼10K	22.1	–	–
(Pourdamghani et al., 2016)	∼10K	26.9	–	–
(Konstas et al., 2017)	∼17K	22.0	–	–
(Song et al., 2016)	∼17K	22.4	–	–
(Song et al., 2017)	∼17K	25.6	–	–
(Flanigan et al., 2016)	∼36K	19.6	–	–
(Pourdamghani et al., 2016)	∼36K	24.3	–	–
(Konstas et al., 2017)	∼20M	33.8	–	–
NMT				
+Delex-Compress-Preorder	∼36K	18.9	26.6	66.2
+Delex+Compress-Preorder		14.6	23.6	77.0
+Delex-Compress+Preorder		19.3	26.3	69.3
+Delex+Compress+Preorder		15.2	23.8	77.8
-Delex-Compress-Preorder		18.2	24.8	67.7
-Delex+Compress-Preorder		15.2	22.4	72.8
-Delex-Compress+Preorder		19.0	25.5	66.6
-Delex+Compress+Preorder		15.9	22.6	71.4
PBMT				
+Delex-Compress-Preorder	∼36K	20.6	32.8	64.5
+Delex+Compress-Preorder		22.2	33.0	63.3
+Delex-Compress+Preorder		24.6	34.3	60.4
+Delex+Compress+Preorder		23.9	33.7	60.5
-Delex-Compress-Preorder		21.0	32.7	65.5
-Delex+Compress-Preorder		25.6	34.1	60.9
-Delex-Compress+Preorder		26.5	34.9	59.9
-Delex+Compress+Preorder		26.8	34.7	59.4

Table 7.1: MT scores for the evaluated models by the size of the training data. Best baseline, PBMT and NMT results were underlined.

Compressing an AMR graph with a classifier shows improvements over a comparable model without compression, but not as strong as preordering the elements in the Linearization step. In fact, preordering seems to be the most important preprocessing step across all three MT preprocessing metrics. We note that the preordering success was expected, based on previous results (Pourdamghani et al., 2016).

Neural MT The first impression from our NMT experiments is that using Compression consistently deteriorates translations according to all metrics evaluated. Delexicalization seems to improve results, corroborating the findings from Konstas et al. (2017). While Delexicalization is harmful and Compression is beneficial for PBMT, we see the opposite in NMT models. Besides the differences between these two MT architectures, applying preordering in the Linearization step improves results in both cases. This seems to contradict the finding in Konstas et al. (2017) regarding neural models. We conjecture that the additional training data used by Konstas et al. (2017) may have decreased the gap between using and not using preordering (see also below). More research is necessary to settle this point.

PBMT vs. NMT PBMT models generate much better sentences from AMRs than NMT models in terms of fluency, adequacy and post-editing effort. We believe that the lower performance of NMT models is due to the small size of the training set (36,521 AMR-sentence pairs). Neural models are known to perform well when trained on much larger data sets, e.g. in the order of millions of entries, as exemplified by Konstas et al. (2017). PBMT models trained on small data sets clearly outperform NMT ones, e.g. Konstas et al. (2017) reported 22.0 BLEU, whereas Pourdamghani

et al. (2016)'s best model achieved 26.9 BLEU, and our best model performs comparably (26.8 BLEU).

Model comparison While the best PBMT models are comparable to the state-of-the-art AMR-to-text systems, the current best results are reported by Konstas et al. (2017), showing the potential of applying deep learning onto large amounts of training data with a 33.8 BLEU-score. However, this result crucially relies on the existence of a very large dataset. Interestingly, when applied in a situation with limited amounts of data, Konstas et al. (2017) report substantially lower performance scores. In such situations, our PBMT models, like Pourdamghani et al. (2016), look appear to be a good alternative option.

7.7 Conclusion

In this work, we systematically studied different MT models to *translate* AMRs into natural language. We observed that the Delexicalization, Compression, and Linearization steps have different impacts on AMR-to-text generation depending on the MT architecture used. We observed that delexicalizing AMRs yields the best results in NMT models, in contrast to PBMT models. On the other hand, for both PBMT models and NMT models, preordering the AMR in Linearization introduces better results.

Among our models, PBMT generally outperforms NMT. Finally, the literature suggests that the improvements obtained by having more data are larger than those obtained with improved preprocessing strategies. Nonetheless, combining the right preprocessing strategy with large volumes of training data should lead to further improvements.

8

Generating text from the semantic web: Comparing modular and end-to-end data driven methods

Abstract In this chapter, we introduce a comparison between modular and end-to-end data-driven approaches to Natural Language Generation (NLG), combining methods explored in the last chapters. In order to have a "common-ground" input representation and a fully unseen test set, we evaluate the different models on how they perform in the WebNLG challenge, which consists of converting non-linguistic data from the Semantic Web into English text. All models were evaluated both automatically and with human judges in an experimental setting. Their implementation is publicly available*.

*https://github.com/ThiagoCF05/CyberTiCC

This chapter is based on Castro Ferreira, T., van der Lee, C., Krahmer, E., & Wubben, S. (2018). Generating text from the Semantic Web: Comparing modular and end-to-end data driven methods. *Manuscript submitted for publication*. An early version was presented in the WebNLG challenge (Gardent et al., 2017a,b).

8.1 Introduction

Natural Language Generation (NLG) is the process of generating natural language from non-linguistic data (Reiter & Dale, 2000; Gatt & Krahmer, 2018). Throughout this thesis, we have studied various aspects of NLG, ranging from, for example, the generation of referring expressions in text to the use of delexicalization in the generation of text from semantic inputs. In this prefinal chapter, we study how these kinds of insights can be combined in full fledged generation systems to convert non-linguistic data from the semantic web into English texts.

Classically, such a full fledged NLG system first has to decide *what* to say, and then choose *how* to say it, steps commonly known as Content Selection and Surface Realization, respectively. To model this process, NLG approaches are often designed in a modular fashion, where Content Selection and Surface Realization are split into several sequential and hierarchical tasks, each one addressed by a separate module. Although these modular systems can perform well on particular domains, they are difficult to develop and maintain and, moreover, adapting them to a different domain can be difficult (Angeli et al., 2010). To solve these issues, researchers have started exploring end-to-end NLG approaches, which are typically developed in a data-driven manner (Belz, 2008; Lu et al., 2009; Wen et al., 2015; Lebret et al., 2016; Chisholm et al., 2017). This naturally raises the question of how both approaches fare, when applied to the same task.

In this chapter, we compare modular and end-to-end data-driven approaches for Surface Realization (i.e., assuming that the content has already been selected). As our modular model, we introduce a system which performs this NLG task in four sequential steps: discourse ordering, template selection, referring expression generation and text reranking. We

will compare the performance of this "classical" NLG set-up with two end-to-end models that convert non-linguistic data into text, adapting Statistical (Koehn et al., 2003) and Neural (Bahdanau et al., 2015) Machine Translation models, respectively.

To decrease data sparsity and account for unseen entities, two of our NLG models, the modular and one end-to-end approach, work by first generating a delexicalized template in which the references are not (yet) textually realized. Once the template is generated by the model, a Referring Expression Generation (REG) module is used to lexicalize the template and produce the final linguistic output.

We evaluated all models in the context of the WebNLG challenge in order to have a single "common-ground" input representation and a fully unseen test set (Gardent et al., 2017a,b). The WebNLG challenge required converting non-linguistic data from the Semantic Web into text. For this goal, a corpus with 25,298 English texts describing 9,674 meaning representations in 15 domains was provided. Based on the outputs of the participating models, an automatic evaluation was performed where the quality of the texts was measured using automatic metrics like BLEU (Papineni et al., 2002), METEOR (Lavie & Agarwal, 2007) and TER (Snover et al., 2006). Additionally, a human evaluation was also performed, in which raters evaluated the quality of a sample of the automatically generated texts, assigning scores for semantics, grammaticality and fluency.

8.2 Related work

As described in Chapter 1, traditional NLG models are often developed using a modular architecture, where Content Selection and Surface Realization are split into several tasks, each one performed by a separate mod-

ule. Practical applications of such modular NLG models can be found in a wide range of domains, including, for example, sportscasting (Theune et al., 2001; van der Lee et al., 2017), weather (Goldberg et al., 1994; Sripada et al., 2004) and pollen forecast (Turner et al., 2006), safety-oriented summaries of scuba dives (Sripada & Gao, 2007), gas turbines event descriptions (Yu et al., 2007), stock market information (Kukich, 1983), personalized smoking cessation letters (Reiter et al., 2003), neonatal intensive care reports (Reiter, 2007; Portet et al., 2009), encyclopedic data (Duma & Klein, 2013; Androutsopoulos et al., 2013) and many others (McKeown, 1982; Iordanskaja et al., 1992).

Although models engineered in a modular structure can perform well on their intended domains, they are difficult to adapt to new ones (Angeli et al., 2010). To address this issue, models which perform NLG in a less modular way have recently been introduced. Such end-to-end approaches normally require a parallel corpus, pairing semantic input with textual output, and make use of a statistical, machine learning model, which is trained to map the non-linguistic source to the natural language target in a more integrated way, using fewer or no intermediate representations.

For example, Belz (2008) introduced a probabilistic grammar to generate forecast text from weather data. The model was trained and evaluated on the SUMTIME corpus (Reiter et al., 2005) and managed to generate forecasts which scored higher than human-produced ones in a human evaluation. Lu et al. (2009) trained and evaluated Tree Conditional Random Fields on the GEOQUERY (Kate et al., 2005) and ROBOCUP (Chen & Mooney, 2008) corpora in order to textually realize geographical queries and soccer statistics, respectively. Wen et al. (2015) proposed a neural generative model with semantically-conditioned Long Short-Term Memory layers (LSTM) to textually realize information about hotel and restaurant

venues. Lebret et al. (2016) and Chisholm et al. (2017) also proposed neural models aiming to generate biographical sentences from fact tables from Wikipedia biographies, whereas the neural model of Dong et al. (2017) textually realizes product reviews.

A fundamental challenge with data-driven NLG models is how to evaluate them. Difficulties have been reported in the direct comparison of different models, even when they are evaluated on the same corpus. Belz et al. (2011), for instance, discussed the case of textually regenerating the Penn Tree bank (PTB) (Marcus et al., 1994). In order to have a non-linguistic input representation for the different models, individual studies typically used parsing trees of the PTB with some (usually lexical) information omitted. The problem, however, is that each study used their own method to process the trees, such that the precise input representations ended up being different across the various studies. Another problem concerns the test part of the evaluation corpus. Once this set is released (together with the rest of the corpus), evaluation results based on it will naturally be reported in the literature. Hence, an idea of the task complexity can be inferred and the data can arguably not be considered fully unseen anymore.

In order to address these issues (lack of a single shared "common-ground" input representation and a truly unseen dataset for a fair evaluation), we compare our modular and end-to-end approaches in the context of the WebNLG Challenge, which involved converting non-linguistic data from the Semantic Web into English text. In the next Section, we describe this challenge in more detail.

Subject	Predicate	Object
Appleton_International_Airport	location	Greenville,_Wisconsin
Greenville,_Wisconsin	isPartOf	Ellington,_Wisconsin
Greenville,_Wisconsin	isPartOf	Menasha_(town),_Wisconsin
Greenville,_Wisconsin	country	United_States
Appleton_International_Airport	cityServed	Appleton,_Wisconsin

The Appleton International Airport is located in Greenville, Wisconsin, United States and serves the city of Appleton, Wisconsin. Greenville is part of the town of Menasha and Ellington, Wisconsin.

Figure 8.1: Example of a set of triples and its respective text.

8.3 The WebNLG challenge

The WebNLG Challenge (Gardent et al., 2017a) consisted of automatically generating English texts to describe non-linguistic data from the Semantic Web. For the challenge, a parallel corpus was provided where the non-linguistic source consists of sets of (up to 7) Resource Description Framework (RDF) triples. These RDF triples offer perhaps the most well known protocol used in the Semantic Web (a *machine-friendly* extension of World Wide Web). Each RDF triple consists of an Agent, a Predicate and a Patient (e.g., `Alan_Bean | occupation | Test_pilot`). The target part of the corpus consists of "crowdsourced" English texts, describing the sets of RDF triples on the source side. Figure 8.1 depicts an example of a set of 5 triples and the corresponding text.

The corpus consists of 25,298 texts describing 9,674 distinct sets of triples in 15 domains: Astronaut, University, Monument, Building, Comics Character, Food, Airport, Sports Team, Written Work, City, Athlete, Artist,

Means of Transportation, Celestial Body and Politician – the last 5 were available only in the test set.

At the beginning of the challenge, only the training and development parts of the corpus – covering 10 domains – were released for training and tuning the participating models. Later, a test set – covering the full 15 domains – was provided, naturally with the source side only. After receiving the test set, the participants had 48 hours to automatically generate the texts to generate descriptions of the source side using their model(s). Based on these texts, the organizers of the challenge conducted both an automatic and human evaluation.

In the following sections, we describe the preprocessing methods used by two of our models for accounting data sparsity and unseen entities, followed by the description of the three models we submitted to the challenge.

8.4 Data preprocessing

In order to decrease data sparsity and account for unseen entities, two of our NLG models work by generating a "template" in which the references are *delexicalized* (e.g., "The main ingredients of AGENT-1 are PATIENT-1."). During the generation process, once the template is produced, their references are lexicalized by a Referring Expression Generation (REG) model.

In the next sections, we explain how we obtained the collection of delexicalized templates by a process called *Delexicalization*, and also how we collected a referring expression dataset to train the model used to lexicalize these templates.

Tag	Entity
AGENT-1	Appleton_International_Airport
BRIDGE-1	Greenville,_Wisconsin
PATIENT-1	United_States
PATIENT-2	Appleton,_Wisconsin
PATIENT-3	Menasha_(town),_Wisconsin
PATIENT-4	Ellington,_Wisconsin

AGENT-1 is located in **BRIDGE-1** , **PATIENT-1** and serves the city of **PATIENT-2** . **BRIDGE-1** is part of **PATIENT-3** and **PATIENT-4** .

Figure 8.2: Mapping between tags and entities and the resulting template.

8.4.1 Delexicalization

Delexicalization is a preprocessing method which aims to decrease data sparsity and account for unseen entities. First, the process automatically maps each entity in a triple set to a general tag: all entities that appear on the left and right side of the triples are respectively mapped to AGENTs and PATIENTs, whereas the entities that appear on both sides are mapped to BRIDGEs. To distinguish different AGENTs, PATIENTs and BRIDGEs in a set, an ordinary ID is given to each entity of each kind (PATIENT-1, PATIENT-2, etc.). Once the entities are mapped on the source side, their referring expressions on the target texts are replaced by the correspondent general tags as Figure 8.2 shows for the example in Figure 8.1.

We used a number of subsequent methods to replace the original referring expressions in the target texts for the respective tags. All the English texts in the training part of the corpus which describe sets of up to 3 RDF triples were manually delexicalized. For the remaining target texts in the training and development sets, we implemented 3 automatic methods to

delexicalize nominal referring expressions, and 1 to delexicalize pronominal referring expressions.

In the first automatic nominal method, we used a list of nominal referring expressions, obtained in the manual delexicalization phase, for each entity on the source side of the corpus. The referring expressions in the target text which matched with one item from the list were replaced by the general tag related to the corresponding entity. In our example in Figure 8.1, if the referring expression *the town of Menasha* was on the list for the entity *Menasha_(town),_Wisconsin*, this referring expression would be replaced in the text for the general tag related to the entity (PATIENT-3).

Next, the second automatic method used a list of referring expressions consisting of Wikipedia IDs of the entities in the source side, in which the underscores were replaced by whitespaces (e.g., *Appleton_International_Airport* $\rightarrow$ *Appleton International Airport*). As in the first method, the other referring expressions in a target text which matched with one in the list were replaced by the general tag relating to the correspondent entity. The determiners which could possibly precede the referring expressions were also replaced. In our example in Figure 8.1, the referring expressions *"The Appleton International Airport"*, *"Greenville, Wisconsin"*, *"United States"*, *"Appleton, Wisconsin"* and *"Ellington, Wisconsin"* would be respectively delexicalized into the general tags of the entities *"Appleton_International_Airport"*, *"Greenville,_Wisconsin"*, *"United_States"*, *"Appleton,_Wisconsin"* and *"Ellington,_Wisconsin"*, as depicted in Figure 8.2.

The third automatic method was similar to the second and also used a list of "Wikipedia" referring expressions. However, instead of exactly matching similar referring expressions, we matched each remaining referring expression in the text with the one in the list considering the shortest

String Edit Distance (Levenshtein, 1966), allowing for approximate string matching.

Finally, to deal with pronominal references we used the Stanford Coreference Tool (Manning et al., 2014) to find all nominal coreferences related to each pronoun in a target text. Subsequently, each pronoun was matched with the entity with the shortest average string edit distance between the Wikipedia ID of the entity and the nominal coreferences.

8.4.2 Referring expression collection

In order to train REG models to lexicalize the templates, we created a dataset with the referring expressions automatically extracted during the delexicalization process. Each extracted referring expression was annotated with their target entity, referential form, syntactic position, discourse and sentence information statuses. Regarding referential form, referring expressions were labeled as pronouns, proper names, descriptions and demonstratives. Referring expressions which started with a determiner article (*the*, *a* and *an*) were labeled as descriptions, and the ones which started with a determiner like *this*, *that*, *these* or *those* where labeled as demonstratives.

Syntactic position was automatically annotated using a dependency tree obtained by the Stanford parser (Manning et al., 2014). As in Castro Ferreira et al. (2016b) (Chapter 3), we classified the syntactic position of a reference as the subject, object or a subject determiner of the sentence. Discourse and sentence statuses represented whether a referring expression was a first or subsequent reference to a given entity in the text or sentence, respectively.

8.5 Models

We introduce one modular (*Pipeline*) and two end-to-end NLG approaches (*SMT* and *NMT*) to convert RDF triple sets into English text.

8.5.1 Pipeline

Our pipeline system produces a text that describes a set of triples in 4 sequential steps: discourse ordering, template selection, referring expression generation and text reranking. Below we offer a brief technical description of each one.

Discourse Ordering is similar to the NLG task of Text Structuring described in Chapter 1 (Section 1.1.1), and aims to find the most likely order(s) of a set of arguments in the discourse by sorting the respective set of triples. The process, sketched in Algorithm 1, relies on two maximum entropy classifiers (ϕ_1 and ϕ_2). The first estimates the likelihood of each triple being the first argument (lines 1-8), using the predicate of the triple and the domain category as features (function f_1). The second classifier estimates the likelihood of the remaining triples being the next argument (lines 9-24). As features (function f_2), it uses the predicates of the target and previous triples, the domain category and a variable which represents whether both involved triples share the same subject or not. The ordering process works iteratively, beam searching the 5 most likely orders of a set of triples (lines 8 and 22).

Template Selection is a step similar to the Lexicalization task described in Chapter 1 (Section 1.1.1). From the training set of manually delexicalized templates described in Section 8.4, our method beam searches the

Algorithm 1 Discourse Ordering Pseudocode

Require: $triples, domain$
1: $ordSets \leftarrow \emptyset$
2: **for all** $triple \in triples$ **do**
3: $features_1 \leftarrow f_1(triple, domain)$
4: $prob_1 \leftarrow \phi_1(features_1)$
5: $candidate \leftarrow \langle triple, prob_1 \rangle$
6: $ordSets \leftarrow ordSets \cup candidate$
7: **end for**
8: $ordSets \leftarrow sortByProb(ordSets)[0, 5)$
9: $i \leftarrow |triples|$
10: **while** $i > 0$ **do**
11: $candidates \leftarrow \emptyset$
12: **for all** $ordSet \in ordSets$ **do**
13: $last \leftarrow lastTriple(ordSet)$
14: $ftriples \leftarrow \{t \mid t \in triples \cap t \notin ordSet\}$
15: **for all** $triple \in ftriples$ **do**
16: $features_2 \leftarrow f_2(triple, last, domain)$
17: $prob_2 \leftarrow \phi_2(features_2)$
18: $candidate \leftarrow ordSet \cup \langle triple, prob_2 \rangle$
19: $candidates \leftarrow candidates \cup candidate$
20: **end for**
21: **end for**
22: $ordSets \leftarrow sortByProb(candidates)[0, 5)$
23: $i \leftarrow i - 1$
24: **end while**
25: **return** $ordSets$

100 most likely templates that describe a given ordered set, as Algorithm 2 depicts. The model looks for the most frequent templates that describe the predicates of the set (lines 5-7). The search (method *search* at line 6) is first carried out among templates of the same semantic category of the triple set, and only considers all the templates in the training set if no template is found in the first attempt. In case no template is found still, the set is split and templates are selected for each subset (lines 8-13).

Referring Expression Generation was performed by a two-step model trained on the training part of the referring expression collection described at Section 8.4. The first step of the model consisted of choosing whether a reference should be a proper name (*Appleton International Airport*), a description (*The airport*), a demonstrative (*This airport*) or a pronoun (*It*). For this choice, we used the Naive Bayes model introduced in Castro Ferreira et al. (2016b) (Chapter 3), trained on the VaREG corpus (Chapter 2; Castro Ferreira et al., 2016a).

Once the referential form was determined, we chose the most frequent form variant in the referring expression collection. Besides the entity to be referred to, the choice for the form variant was conditioned on features like syntactic position and information status in the discourse and the sentence. If a referring expression is not found, we realized the reference using the Wikipedia ID of the entity, replacing each underscore in the ID for whitespaces (e.g., Appleton_International_Airport to "Appleton International Airport").

Text Reranking orders the 100 most likely texts that describe the input set of triples using a 6-gram language model trained on the Gigaword Corpus Third Edition with KenLM (Heafield et al., 2013). The highest

Algorithm 2 Template Selection Pseudocode

Require: $ordSet, domain$

$begin \leftarrow 0$
$end \leftarrow |triples|$
$templates \leftarrow \emptyset$
while $begin < |ordSet|$ **do**
    $predicates \leftarrow getPredicates(ordSet[begin, end))$
    $candidates \leftarrow search(predicates, domain)$
    $subTemps \leftarrow sortByProb(candidates)[0, 100)$
    if $|subTemps| = 0$ **then**
        $end \leftarrow end - 1$
        if $begin = end$ **then**
            $begin \leftarrow begin + 1$
            $end \leftarrow |triples|$
        end if
    else
        if $|templates| = 0$ **then**
            $templates \leftarrow subTemps$
        else
            $cands \leftarrow \emptyset$ ▷ Candidate templates
            for all $template \in templates$ **do**
                for all $subTemp \in subTemps$ **do**
                    $temp \leftarrow template \cup subTemp$
                    $cands \leftarrow cands \cup temp$
                end for
            end for
            $templates \leftarrow sortByProb(cands)$
            $templates \leftarrow templates[0, 100)$
        end if
        $begin \leftarrow end$
        $end \leftarrow |ordSet|$
    end if
end while
return $templates$

scoring text according to the language model is returned as the one that best describes the triple set.

8.5.2 SMT

SMT is a Phrase-based Machine Translation model built on the Moses toolkit (Koehn et al., 2007). It aims to *translate* a linearized set of triples into an English text. For the example in Figure 8.1, a linearized version of its set would be:

Appleton_International_Airport location Greenville,_Wisconsin
Greenville,_Wisconsin isPartOf Ellington,_Wisconsin
Greenville,_Wisconsin isPartOf Menasha_(town),_Wisconsin
Greenville,_Wisconsin country United_States
Appleton_International_Airport cityServed Appleton,_Wisconsin

Based on the poor performance of Statistical Machine translation systems in predicting delexicalized templates reported in Castro Ferreira et al. (2017a) (Chapter 7), delexicalization was not used in this approach. Instead, the model was trained on the lexicalized version of the WebNLG training set, augmented with a group of reference pairs to improve the generation of referring expressions. Each reference pair consisted of Wikipedia entities in the source side (e.g., Greenville,_Wisconsin), and referring expressions in the target side (e.g., Greenville). The pairs were extracted based on the referring expressions obtained in the manual delexicalization phase described at Section 8.4.

Most of the model settings were copied from our Statistical MT system for AMR-to-text (Chapter 7; Castro Ferreira et al., 2017a). At training time, we extracted and scored phrases up to the size of 20 tokens. As feature functions, we used direct and inverse phrase translation probabilities

and lexical weighting, as well as word, unknown word and phrase penalties. These feature functions were trained using alignments from the training set obtained by MGIZA (Gao & Vogel, 2008). Model weights were tuned on the development data using 60-batch MIRA (Cherry & Foster, 2012) with BLEU as the evaluation metric. A distortion limit of 6 was used for the reordering models. We used two lexicalized reordering models: a phrase-level (phrase-msd-bidirectional-fe) (Koehn et al., 2005) and a hierarchical-level one (hier-mslr-bidirectional-fe) (Galley & Manning, 2008). At decoding time, we used a stack size of 1000. The language model was also a 6-gram LM trained on the Gigaword Third Edition corpus using KenLM.

8.5.3 NMT

NMT is a neural model based on the Edinburgh Neural MT submission (UEDIN-NMT) for the shared translation task at the 2016 Workshops on Statistical Machine Translation (Sennrich et al., 2016a) [†]. This model aims to predict a template (with a maximum sentence length of 50) for describing a linearized and delexicalized set of triples. For the example in Figure 8.1, a linearized and delexicalized version of the set would be:

SUBJECT-1 location BRIDGE-1 BRIDGE-1 isPartOf OBJECT-4
BRIDGE-1 isPartOf OBJECT-3 BRIDGE-1 country OBJECT-1
SUBJECT-1 cityServed OBJECT-2

In order to have an open vocabulary, we split rare tokens on the source and target sides in sub-word units using Byte Pair Encoding (BPE) (Sennrich et al., 2016b). Source and target word embeddings were 620D each,

[†] https://github.com/rsennrich/wmt16-scripts

whereas hidden units were 1024D. Gradients were normalized to 1.0. Models were trained using stochastic gradient descent with Adadelta (Zeiler, 2012) and mini-batches of size 80. We applied early stopping for model selection based on BLEU scores (20 epochs), and dropout with a probability of 0.1 in both source and target word embeddings and 0.2 for hidden units. Decoding was performed with a beam search of size 12. In general, we did not work on finding the optimal parameter settings, but mostly relied on default settings. In a few cases, we adjusted the default settings if prior experiences suggested this would be helpful.

Once the template was predicted, we used the same Referring Expression Generation module of *Pipeline* to lexicalize the template.

8.6 Evaluation

We compared our models based on their results in the WebNLG Challenge (Gardent et al., 2017a), where automatic and human evaluations were performed. In the automatic evaluation, BLEU (Papineni et al., 2002) (up to 3 references), METEOR (Lavie & Agarwal, 2007) and TER (Snover et al., 2006) were computed and statistically tested using the bootstrapping algorithm of Koehn & Monz (2006). Higher values for BLEU and METEOR represent better performance of the models, whereas for TER the opposite holds.

In the human evaluation, crowdworkers were recruited to rate the texts produced by each participating model for describing 223 triple sets sampled from the test part of the corpus. Using a 3-point Likert scale (1 - Bad; 2 - Medium; 3 - Good), participants rated the semantics (*does the text correctly represent the meaning in the data?*), grammaticality (*is the text grammatical (no spelling or grammatical errors)?*) and fluency (*does*

		BLEU	METEOR	TER
All	SMT	**44.28**	**0.38**	**0.53**
	NMT	34.60	0.34	0.60
	Pipeline	35.29	0.30	0.56
Seen	SMT	**54.29**	**0.42**	**0.47**
	NMT	43.28	0.38	0.51
	Pipeline	44.34	0.38	0.48
Unseen	SMT	**29.88**	**0.33**	**0.61**
	NMT	25.12	0.31	0.72
	Pipeline	20.65	0.21	0.65

Table 8.1: Automatic evaluation results on all domains as well as on domains only seen during training and development sets and unseen domains. Based on the statistical tests, results ranked first are written in bold face, whereas the ones ranked second are underlined.

the text sound fluent and natural?) of the automatically generated texts. A Wilcoxon rank-sum test was performed in order to test the statistical significance of the results. More information about the human evaluation can be found in the final report of the WebNLG challenge[‡].

8.7 Results

8.7.1 Automatic evaluation

Table 8.1 depicts the BLEU, METEOR and TER results of our three models on all domains, domains seen on training and development sets, and unseen domains (only present on test set). On all domains, our Statistical Machine Translation model (*SMT*) obtained results significantly better

[‡]http://webnlg.loria.fr/pages/webnlg-human-evaluation-results.pdf

		Semantics	Grammar	Fluency
All	SMT	1.96	**2.42**	1.81
	NMT	2.16	1.99	2.01
	Pipeline	**2.19**	2.20	**2.07**
Seen	SMT	2.14	**2.47**	2.01
	NMT	2.23	**2.14**	2.10
	Pipeline	**2.33**	2.46	**2.21**
Unseen	SMT	1.61	**2.30**	1.44
	NMT	**2.01**	1.69	**1.85**
	Pipeline	1.91	1.70	1.78

Table 8.2: Human evaluation results for the three models on all domains as well as on domains only seen on training and development sets and unseen domains. Results ranked first are written in bold face, whereas the ones ranked second are underlined.

than the two other models (*Pipeline* and *NMT*). When comparing *Pipeline* and *NMT*, we can observe that the models scored significantly different in terms of BLEU. Additionally, *NMT* outperformed *Pipeline* in terms of METEOR, whereas the latter scored significantly higher (i.e., worse) TER scores than the former.

Also when looking at results for seen and unseen domains, we see that *SMT* similarly obtained the best results, significantly outperforming the other two models. Between *NMT* and *Pipeline*, the results on seen domains were not significantly different in terms of METEOR, whereas the former outperformed the latter on unseen domains. In terms of TER, *Pipeline* performed significantly better than *NMT*.

8.7.2 Human evaluation

Table 8.2 summarises the results for semantics, grammaticality and fluency of the models on all domains, domains seen on training and development sets, and unseen domains. Inspection of this table reveals a different, more complicated pattern than the results of the automatic measures. Texts produced by the *Pipeline* were the ones rated as most fluent and as best describing the input data on all domains as well as on domains only seen during training and development. On unseen domains, *NMT* was the model that yielded the best results for both these metrics, with *Pipeline* as the runner up. On the other hand, similar to the automatic evaluation, human judges systematically scored *SMT* as the best model in terms of grammaticality.

In sum, the texts produced by *SMT* were the ones with least spelling and grammatical errors, but they were presumably not as fluent and did not describe the meaning of the input data as well as *Pipeline* on seen domains and *NMT* on unseen ones.

8.8 Discussion

In this prefinal chapter, we integrated a number of the findings described earlier in this thesis into comprehensive NLG systems, capable of generating natural language output based on non-linguistic semantic input representations, derived from the the Semantic Web (RDF triples). In doing so, we looked at both modular and more integrated, data-driven end-to-end NLG approaches. Our modular model, called *Pipeline*, generated a text from a set of RDF triples in 4 sequential steps: discourse ordering, template selection, referring expression generation and text reranking. Our two end-to-end approaches, in contrast, did not rely on these four interme-

diate steps, being more integrated. Specifically, the first end-to-end model, called *SMT*, is a phrase-based machine translation method that converts a linearized set of triples into English text, whereas the second, called *NMT*, is a Neural Machine Translation model which converted a linearized and delexicalized set of triples into a template which was then lexicalized with a Referring Expression Generation model.

The performance of all three systems was compared based on their results in the automatic and human evaluations performed on the WebNLG Challenge, to which 6 other systems were submitted. We found that in the automatic evaluation the *SMT* system systematically outperformed our other two approaches and ranked among the top systems in the overall ranking, scoring the 2nd position in terms of BLEU, METEOR and TER. When looking at the human evaluation, however, the *SMT* system performed well in terms of Grammaticality (3rd position in the overall ranking), but not as good in terms of Semantics and Fluency, scoring the 7th and 8th position, respectively. The *Pipeline* and *NMT* systems, by comparisons, ranked 4th and 5th on these two dimensions, respectively. Overall, the best system in the automatic evaluation was a neural machine translation approach submitted by the University of Melbourne, Australia, whereas in the human evaluation, the best results were obtained by a modular template-based approach submitted by the Universitat Pompeu Fabra, Barcelona, Spain. More information about the challenge can be found in the official website[§]. Below we discuss a number of aspects of our experiences in this chapter in somewhat more detail.

Automatic vs. Human Evaluation The results of the automatic evaluation did not correlate with those of the human evaluation for most of

[§]http://webnlg.loria.fr/pages/challenge.html

the metrics. For instance, BLEU, which is an automatic metric often used to automatically estimate the fluency of a generated text, scored the texts produced by our Phrase-based MT model (*SMT*) the highest, whereas the same texts were assessed by human judges as the least fluent ones of the three. As it turned out, grammaticality was the only measure in the human evaluation which correlated with the automatic metrics, depicting *SMT* as the best performing model. On the other dimensions assessed in the human evaluation, *Pipeline* scored highest for semantics and fluency on domains seen during the training process, whereas *NMT* obtained somewhat better scores for both metrics on unseen domains.

The fact that automatic metrics like BLEU and human assessments did not correlate is consistent with earlier discussions in the literature (Novikova et al., 2017a) and shows the need for automatic measures to better evaluate automatically generated natural language. Therefore, since human judges can be assumed to comprehend and assess natural language better than machines so far, we primarily refer to differences between the models based on the results of the human evaluation instead the automatic one.

Modular Approach We introduced a modular approach, called *Pipeline*, which, like most traditional NLG models, tackles the problem of generating text based on semantic input by relying on several sequential and hierarchical steps (Reiter & Dale, 2000). According to the human evaluation, this model generated more natural and fluent texts, which better describe the meaning of the input data than the end-to-end approaches on already seen domains. However, the results of this model for both metrics presented a considerable drop on unseen domains, confirming the difficulties reported on adapting modular NLG systems to different domains (Angeli

et al., 2010).

End-to-end Approaches Our end-to-end approaches were based on Machine Translation methods. Of the three models introduced in this chapter, the end-to-end approach based on Statistical Machine Translation, *SMT*, was the one that produced the texts with the least spelling and grammatical errors. However, this approach was rated the lowest in terms of the fluency of the generated texts, as well as on how well they represented the meaning of the input data.

Although the texts produced by our end-to-end approach based on Neural Machine Translation (*NMT*) contained more linguistic errors, humans judges scored them higher in terms of fluency and semantics than texts produced by *SMT*. The fluency and semantics of this model also did not present such a substantial drop on unseen domains, being a better option to be generalized to new domains than the modular approach.

Delexicalization Two of our models, the modular approach *Pipeline* and the end-to-end approach *NMT*, first generated a delexicalized template in which the referring expressions were not textually realized. Next, a REG model was used to lexicalize the produced templates. In the results of the human evaluation, we noticed that both models managed to generate more fluent texts, which better describe the input data, but with more linguistic errors than the model which directly maps a non-linguistic input to an English text (*SMT*). On balance, we interpret this as a tentative evidence of the benefits of using delexicalization for this task.

Conclusion In this chapter, we developed one modular approach and two end-to-end approaches to convert non-linguistic data from the Semantic Web into English text, inspired by the earlier chapters of this thesis.

According to the results, the *SMT* approach, which directly maps the non-linguistic representation onto a linguistic output, produced texts with least spelling and grammatical errors. On the other hand, our modular and end-to-end approaches, making use of delexicalization, generated the most fluent texts, that best described the data. In the comparison among them, the modular approach performs better on seen domains, whereas our end-to-end approach making use of delexicalization seems to be better adapted to new domains.

9

General discussion and conclusion

In this thesis, we presented a number of studies in the field of Natural Language Generation (NLG), which is the process of automatically converting semantic input data into coherent, natural language output text. Since we already discussed the specific findings in the individual chapters, here we highlight a number of more general topics, focusing on issues related to modeling referential variation (Section 9.1), the nature of semantic input representations (Section 9.2), the comparison between modular and integrated, end-to-end NLG approaches (Section 9.3) and issues in the evaluation of NLG systems in general (Section 9.4). We end with a number of pointers for future research in Section 9.5.

9.1 Modeling variation

In Chapters 2 to 6, we studied linguistic variation in the process of automatic text generation, focusing on models of Referring Expression Generation (REG). By developing REG models able to generate more varied noun phrase references, we hope to be able to increase the *humanlikeness* of generated texts, since human authors typically are capable of generating varied texts to express the same communicate goal, as previously shown in literature and confirmed by our corpus analyses. In this thesis we have looked at two kinds of variation, which we can refer to as pragmatic and individual variation. We briefly discuss both below.

9.1.1 Pragmatic variation

With pragmatic variation, we refer to how the form and content of referring expressions may vary as a function of the context in which they occur, which in our case mainly consisted of the discourse context. We studied how the discourse salience of a referent influenced the choice of referential form as well as the specific proper name form. According to previous psycholinguistic studies (e.g., Gundel et al., 1993; Grosz et al., 1995; Jaeger, 2010), salient references in discourse are more likely to be referred to using shorter referring expressions (like pronouns) than less salient ones, which are typically referred to using longer expressions (like full proper names). To account for what causes a reference to be salient in discourse, we used features which had already been used to distinguish salient and non-salient references, like syntactic position (Brennan, 1995), referential status and recency (Chafe, 1994).

Choice of Referential Form To study variation in referential form, we first collected and analyzed a new corpus, which we call VaREG (Chapter 2). To collect this dataset, we presented different writers with texts in which all references to the main topic were replaced with gaps. Participants were asked to fill each gap with a reference to the topic. This resulted in 9,588 referring expressions, produced by 78 different participants for 563 referential gaps. In our analysis of the VaREG corpus, we indeed found that modeling the discourse context based on a notion of salience helps to explain the pragmatic variation in the choice of referential form. We noticed that writers were more likely to use proper names for less salient referents – e.g., initial references in discourse (91% of the choices) – and to use pronouns for more salient ones – e.g., subsequent references in sentence (76% of the choices). Based on these findings, we decided to computationally model the choice of referential form selection in Chapter 3 using discourse features like syntactic position, referential status and recency.

Proper Name Generation To study variation in proper names, we collected another dataset, which we dubbed REGnames (Chapter 4). It is a corpus consisting of 53,102 proper names references to 1,000 different persons in more than 15,000 webpages extracted from the Wikilinks corpus. When looking at which factors influence the form of proper names, we found comparable effects of salience: longer proper names (both in terms of number of tokens and number of proper name attributes) were more likely to be used for less salient referents, such as initial ones, mentioned early in the text or in the object role of a sentence. In a similar vein, the average length of proper name references decreased as a function of the sentence rank in the discourse, when referents presumably become more

salient due to the increased number of previous references in the text.

In our analyses of the REGnames corpus, we also found an additional factor that influenced how proper name forms are realized in text besides discourse factors: the specific entity mentioned turns out to also have a big influence on the form of a proper name. For example, in a same discourse context, the combination of first and last birth names might count as a full name for some people (like *Marisa de Azevedo Monte*), whereas for others, this would hold for the combination of first and middle birth names (like *Elis Regina Carvalho Costa*).

Based on these findings, our models for proper name generation (described in Chapter 5) take pragmatic variation into account by relying on discourse features like syntactic position and referential status, as well as the target entity itself.

End-to-End REG The approach to REG described in Chapters 3 and 5 relied on a small set of specific features, as just mentioned. Additional features could have been included. For instance, in the VaREG corpus (Chapter 3) we saw that there was a significant difference in the choice of referential forms as a function of discourse genre (news, product review or encyclopedia text). Moreover, according to previous studies, other features could also be used to model the salience of a referent, such as topicality and parallelism (Arnold, 1998). However, many of these features are not readily available or are difficult to extract automatically, in contrast to the features we did use in our models of reference (and which already yielded remarkably good results).

In short, feature engineering is a complex task. In order to avoid it, Chapter 6 introduced a novel end-to-end REG approach called NeuralREG, which, in contrast to traditional approaches, generates referring ex-

pressions to discourse entities in text by simultaneously selecting form and content without any need of explicitly extracting discourse features like syntactic position, referential status or recency. Instead, the model takes pragmatic variation into account by first encoding the surrounding linguistic contexts and combining them into single vector representations, which are subsequently used to produce a referring expression to the target, suitable for the specific discourse context.

9.1.2 Individual variation

If pragmatic variation focused on how referring expressions vary as a function of different discourse contexts, individual variation looks into the varied ways in which speakers or writers could vary a particular noun phrase in the same (or a very similar) context. We took different approaches to study this phenomenon in the choice of referential form and proper name generation, respectively.

Choice of Referential Form The VaREG corpus consists of around 20 referring expressions produced by different writers for each referential gap, in various discourse contexts. These multiple gold-standards allowed us to conduct a detailed analysis of the agreement between writers in choosing the form of a reference in the same situation. Our analyses revealed that there is substantial individual variation in the choice of referential form between different authors. Interestingly, salience seems to partly explain the amount of individual variation that can be observed. For example, we noticed a higher amount of this kind of variation when writers had to choose referential forms for the direct object position of a sentence as well as for references that were relatively distant from the most recent previous reference to the same topic.

Relying on our findings for the VaREG corpus, we developed a Naive Bayes and a Recurrent Neural Network (RNN) model for the choice of referential form in Chapter 3. Both models take individual variation into account by predicting a frequency distribution over all referential forms instead of a single one. The models were evaluated on the VaREG corpus, comparing the predicted frequency distributions with the gold-standard ones, relying on syntactic position, referential status and recency features to model the discourse context (pragmatic variation).

In an automatic evaluation, the Naive Bayes model, trained on the VaREG corpus, performed best in modeling the individual variation in the choice of referential form. Even though RNNs model the selection of referential forms for a target reference based on the forms of the previous references, they did not perform better than the Naive Bayes model, which does not take the history of references into account. We believe that the lower performance of RNNs can be attributed, at least in part, to the size of the training corpus, since RNNs typically need a large data set to be trained on. Moreover, we also conjecture that the referential status feature, used by both models, might have been sufficient to model the relation between a reference and its antecedent(s), favoring the simpler Naive Bayes model.

Besides the automatic evaluation, we also performed a human evaluation, where we used our best performing model as part of a full-blown REG model for generating varied referring expressions to the topic of texts from the GREC-2.0 corpus (Belz et al., 2010). Our model worked by first grouping the references in a similar discourse context according to their feature values. Then the frequency distribution over the referential forms is applied to this group of references in such a way that their forms are representative of the predicted distribution. For instance, if a frequency distribution of 0.8 proper names and 0.2 pronouns is predicted for a group

of 5 references in a similar discourse context, 4 of these references would be realized as proper names and 1 as a pronoun.

We evaluated the coherence and comprehensibility of the resulting texts in comparison with both a version in which references were produced by a random baseline model and the original texts. We found that the texts in which the references had their forms generated by our model were not rated significantly different from the original texts, and both were judged as significantly more coherent than the texts with randomly generated references. This is an indication that our solution does not only nicely model individual variation in the choice of referential form, but also that this does not negatively affect the quality of the output texts. This is an important step towards new models for automatic text generation that are less predictable and more varied.

Proper Name Generation Different from VaREG, the corpus we collected for the study of proper names (REGnames, Chapter 4) does not have multiple gold standard referring expressions for the exact same referential context and, for this reason, we could not do a detailed analysis of the individual variation for the proper name generation task. Still, we did develop a proper name generation model in Chapter 5 which could generate varied forms in a similar manner as the models in Chapter 3. That is, this variant of the model would first predict a frequency distribution over all proper name forms for the similar references in the discourse, and select forms in accordance with the overall distribution.

In a human evaluation, we compared the effects of this model with an alternative that did not attempt to model variation in this way. Both of these models outperformed a number of competing systems for proper name generation we implemented, based on proposals by van Deemter

(2016) and Siddharthan et al. (2011). However, we also found that texts with references generated by our "no individual variation" model were preferred by judges over texts that included non-deterministically generated, varied proper name references. This result suggests a preference for consistency in proper name references in similar situations, which appears to be different from the choice of referential form.

End-to-End REG Even though we did not test this explicitly, we hypothetically assume that our end-to-end REG model, NeuralREG, can also take individual variation into account, since, in its decoding step, it can beam search a group of varied referring expression candidates that are likely to suit a given discourse context. It would be interesting to explore this in more detail in future research.

9.2 Semantic representations

As described in the Introduction (Chapter 1), there is no consensus on what the input representation of an NLG system should be. NLG systems have been developed for a wide range of different input representations, including images, numeric data and (other) meaning representations. This lack of a "common" input representation is a limitation for comparing NLG systems and for exchanging insights and technical implementations between them. To address this problem, researchers have started looking for candidate input formats that could be used more broadly within the community. In this thesis, we have looked in detail at two of them: Abstract Meaning Representation (Chapter 7) and RDF Triples from the semantic web (Chapter 8), which have fundamental differences in terms of level of specification, limitations and availability of resources.

Abstract Meaning Representations (AMRs) are structures that encode the meaning of sentences as rooted, directed and acyclic graphs, where nodes represent concepts, and labeled directed edges represent relations between these concepts. Entities are normally represented by their semantic typing and Wikipedia IDs (wikified references). In general, AMRs are meaning structures which are relatively close to their final linguistic realization, where the meaning representation has already been structured in sentences containing some syntactic and lexical information (e.g., verb framesets and function words). But even with this high specification level, AMRs miss important information which would be useful for the generation process, such as information on number agreement as well as coreferences and rhetoric information between sentences.

RDF Triples are representations which are specified in less detail, and which are also not so close to the intended linguistic realization as AMRs. Each RDF unit consists of two entities, a Subject and an Object, both represented by their Wikipedia IDs (or constants), and related by a predicate. No information about sentence ordering, nor about syntactic or lexical information is represented in these structures. Moreover, RDFs do not contain temporal information, which implies that determining the tense of verb phrases during the generation process is a challenge. In addition, it is difficult to represent multiple connections among concepts, since in its current format, each RDF unit only expresses a relation between two entities.

Generation from AMRs and RDF triples In Chapters 7 and 8 we studied generation from AMRs and RDF triples, respectively. We used comparable approaches to convert both semantic representations into English texts. For AMRs, we developed various Statistical and Neural Machine

Translation models, and applied them to different preprocessed versions of AMRs. For RDF triples, we used Machine Translation models comparable to the ones used for AMR-to-text, but we also developed a modular NLG model which generates an English text from a set of RDF triples in 4 sequential steps. A direct comparison is not possible, because the systems were trained on different corpora, and generated texts of different levels of complexity. In general, we did find that the texts generated by the RDF-to-text models seemed to result in higher average fluency, adequacy and post-editing scores than the texts produced by the AMR-to-text models, which we primarily attribute to the lower complexity of the generated sentences.

Concluding remarks In general, both RDF triples and AMRs are helpful formats for NLG research, and which is preferred presumably depends on the specific goal of the NLG system to be developed or on the NLG problem to be addressed. For instance, to study the full textual realization process, working with RDF triples seems preferable over AMRs, while for text-to-text NLG approaches or for the study of specific issues, such as lexical choice or phrase ordering within a sentence, AMRs may be the better choice. An important factor will also be for which representation most resources will be available, and here the size of the Semantic Web may provide an argument in favour of RDF triples. On the other hand, although there are not that many AMR resources available, various parsers have been developed to automatically extract these meaning representations from text (Flanigan et al., 2014; Artzi et al., 2015).

9.3 Modular vs. end-to-end approaches

Both for REG (Chapters 2-6) as well as for the generation of text from meaning representations (Chapter 7-8), we studied modular and end-to-end data-driven approaches. The former typically generate natural language in a pipeline architecture, where several subtasks are performed by different modules in a cascade style, resulting in the final linguistic output; the latter, by contrast, tackle the problem in a single, integrated and less modular framework. Below we discuss our experiences for both tasks.

Referring Expression Generation is arguably a specific subtask in modular approaches for NLG in itself (Reiter & Dale, 2000). We first approached this task using a two module REG architecture, where first the form of a reference is chosen, after which it is decided how to realize the selected form. As discussed above, in Section 9.1, we developed data-driven models for both subtasks: the choice of referential form and proper name generation. Although the models yielded a good performance, the process of engineering features to model the context as well as the integration of the different modules can be highlighted as two issues with this approach.

As an alternative to the modular REG architecture, we introduced an end-to-end approach based on neural networks which tackles the full Referring Expression Generation process, producing references to discourse entities in text, simultaneously selecting form and content without any need for feature extraction techniques. In an automatic evaluation, the 3 variants of this model (plain sequence-to-sequence, concatenative- and hierarchical-attention) outperformed the modular REG approach based on Chapter 3. Moreover, in a human evaluation, texts with the referring expressions generated by the variant with a concatenative-attention mecha-

nism were judged significantly more fluent than texts with referring expressions generated by the same modular approach.

Meaning Representation to Text In Chapter 7, we evaluated the performance of modular and end-to-end approaches to generate English texts based on non-linguistic data from the Semantic Web (sets of RDF triples). The modular model converts a set of RDF triples into an English text in 4 sequential steps (discourse ordering, template selection, referring expression generation and text reranking), whereas the end-to-end approaches were based on Machine Translation models which *translate* the input representation into a linguistic output, relying less on intermediate stages or representations. In a human evaluation, we found that the more integrated approach, a Statistical Machine Translation model, managed to generate the more grammatical texts, whereas the modular and end-to-end approaches making use of delexicalization, generated the more fluent texts, that also were judged to better represent the meaning of the non-linguistic input data.

Concluding remarks Both modular and integrated approaches have their strengths and weaknesses when it comes to automatically generating natural language texts. Modular approaches may generate high quality texts, relying less on resources like data and computational power, but they demand more engineering work, given their more complex architectures. On the other hand, end-to-end approaches may demand less engineering work, but require more computational power and larger amounts of data covering different scenarios in order to be trained adequately and to generate coherent and comprehensible texts. In our own experiments, we found that an integrated approach outperformed the modular approach to REG, but the findings for RDF-to-text were less clear cut.

9.4 Evaluation of NLG

Evaluation is an important (and challenging) aspect of any NLG study. In this thesis, we (automatically) evaluated all NLG models that we developed. Moreover, with the exception of the AMR-to-text models described in Chapter 7, the output of all systems has also been evaluated by human evaluators. Below we reflect upon the outcomes of these evaluation exercises, each involving different goals, different metrics and different datasets.

Choice of referential form In Chapter 3, we automatically evaluated the models for the choice of referential form by cross-validating them on our multiple-gold standard VaREG corpus using the Jensen-Shannon divergence metric and the Spearman's rank correlation to measure to which extent they succeeded in modeling the individual variation found for the task. We found that all our new models outperformed a random and a deterministic baseline. Then, based on the results of the automatic evaluation, we selected the best performing model to be compared to original texts in a human evaluation experiment. Participants were asked to rate, on a 5-point Likert-scale, the coherence and comprehensibility of the texts with referring expressions generated by our best model, in contrast to the original texts as well as texts with randomly generated references. We found that the texts with our model's referring expressions were not rated significantly different from the original texts, and significantly better than the texts with random referential forms in terms of coherence.

Proper Name Generation Next, we evaluated the performance of our proper name generation models, described in Chapter 5, using the REG-names corpus (Chapter 4). We used accuracy to measure the models' per-

formance in the prediction of proper name forms (*first+last* names, *last* name, etc.) and string edit distance to measure the quality of the final output in comparison with a gold-standard proper name reference. In a 10-fold cross validation, both measures pointed towards the same model as being the best. Additionally, they suggested that our models all performed better than the comparison baselines. Then, in a human evaluation, we pairwise compared original texts against alternative versions with proper name references generated by our model, with and without accounting for individual variation. The human judges preferred the original texts over the versions relying on our models. Additionally, they preferred texts with proper names generated by our deterministic model over the variant with references generated by our individual variation model. In other words, we did not find a clear benefit of modeling individual variation when it comes to proper name forms.

End-to-end REG Our end-to-end model for REG (NeuralREG) was trained and evaluated on a delexicalized version of the WebNLG corpus (Gardent et al., 2017a,b). In an automatic evaluation, we used accuracy and string edit distance to evaluate the quality of the referring expressions generated by our approach in comparison with gold-standard referring expressions. In contrast to the evaluation of the other REG models in this thesis, we also compared the quality of the texts with NeuralREG's referring expressions with the original texts in the automatic evaluation, measuring accuracy and BLEU (Papineni et al., 2002). All measures indicated a significantly better performance of our model in comparison with the baselines.

In a human evaluation, we asked judges to rate the fluency, grammaticality and clarity, on a 7-point Likert-scale, of the original texts and the

alternative versions with references generated by our proposed models and baselines. The results correlated with those of the automatic evaluation: our proposed models numerically outperformed the baselines and performed very similar to each other in terms of the three metrics, although the differences were only statistically significant for one of the systems and only for one of the metrics (fluency).

Since we observed differences between both kinds of evaluation, we looked in more detail at the possible correlations between the automatic metric BLEU on the one hand, and the human judgments on the other. While we found clear correlations between the various human scores, we did not observe significant correlations between the BLEU scores and the human judgments.

Meaning Representation to Text Both for the RDF-to-text and the AMR-to-text models, the fluency, adequacy and post-editing effort of the generated texts were automatically measured using the BLEU (Papineni et al., 2002), METEOR (Lavie & Agarwal, 2007) and TER (Snover et al., 2006) measures, respectively. In both tasks, the statistical machine translation models obtained the highest scores according to the three automatic metrics.

The texts produced from RDF triples were also evaluated in a human experiment, where human judges rated the quality of the texts according to semantics (does the text correctly represent the meaning in the data?), grammaticality (is the text grammatical (no spelling or grammatical errors)?) and fluency (does the text sound fluent and natural?). According to the results, the statistical machine translation approach produced texts with least spelling and grammatical errors. On the other hand, our modular and neural machine translation approaches, making use of delexical-

ization, generated the most fluent texts, that also described the data best. In the comaprison between both, the former performed better on seen domains while the latter performed better on unseen ones.

Again, the results of the automatic evaluation did not correlate clearly with the results of the human evaluation for most of the metrics. For instance, the texts produced by our statistical machine translation model scored high on the BLEU metric, while the same texts were assessed as the least fluent ones by human judges.

Concluding remarks The result of our evaluation studies are difficult to compare to each other, since they rely on different systems, different datasets, and different tasks and hence also involve different metrics. However, the general patterns we observe are very similar to results of earlier NLG evaluations (Stent et al., 2005; Belz & Reiter, 2006; Novikova et al., 2017a). Most notably, we found that automatic measures like BLEU, METEOR and TER correlate only to a limited extent (and sometimes not at all) with the results obtained from human judges. This lack of correlations has been discussed in the literature before (Dusek et al., 2017; Novikova et al., 2017b), and highlights the need for better automatic metrics. In general, our findings confirm that it is good practice to combine different measures, both automatic and human ones, when evaluating NLG systems.

9.5 Future research

Before we conclude this thesis, we would like to mention four lines for future research.

Beyond discourse context One aim of this thesis was to develop REG models that are capable of generating references in a varied way, incorporating both individual and pragmatic factors. We have shown that using just a few general features, like syntactic position, referential status and recency, can bring us a long way. However, all these features are related to a local discourse context, much as our end-to-end neural REG model, which produces a referring expression conditioned on a single vector representation that encodes only the discourse context and the entity to be referred to. The generation of referring expressions in other kind of contexts, like the visual one, is out of scope. Furthermore, even in the domain of discourse, more global contextual information is not taken into account, and as a result referring expressions like "Senator Barack Obama" or "President Dilma Rousseff" (which may appear in the training data), could mistakenly be generated in the context of when this thesis was written, even though the target entities were a former senator/president and a former president, respectively.

In future research, we would like to study the generation of referring expressions, not only taking pragmatic, discourse variation into account, but also in broader scenarios which may not need always be explicit in the discourse. For instance, we could explore multi-modal approaches to REG, such as the one proposed by Andreas & Klein (2016), which generates rich, contextually appropriate descriptions of structured world representations using neural networks that process the context both visually and textually. Moreover, in our end-to-end approach, we might encode the context not only using word embeddings trained based on local occurrences in text, but also based on information from global databases like the semantic web. By including a more "global" representation of an entity and the context in which it occurs, the decoder, when conditioned to this

representation, would hopefully generate a referring expression according to global constraints (like temporal ones).

Data resources The limitation of data resources has been a frequently discussed topic in the NLG field of research (like in Natural Language Processing more in general) (Novikova et al., 2017b). Recently, several data resources for NLG have been released, such as the AMR dataset (Banarescu et al., 2013), the WebNLG challenge (Gardent et al., 2017a,b) and the E2E task (Novikova et al., 2017c). In general, collecting resources like these is expensive and time consuming, since the creation typically involves manual, possibly crowdsourced labor. For future work, it would be interesting to develop methods to automatically create data resources for NLG, like the generation of parallel synthetic data. For instance, from an aligned parallel corpus pairing AMRs and the related parses of English syntactic trees, the most likely co-occurrences of meaning subgraphs and syntactic subtrees might be combined to form novel parallel instances. In a similar vein, it would be worthwhile to improve parsers which work in the opposite direction of generation models, extracting the meaning representation from text like Flanigan et al. (2014) and Artzi et al. (2015). In general, it is to be expected that the importance of data for NLG will continue to increase in the coming years, and new ways of collecting and possibly generating data resources should be explored.

Evaluation of NLG Our evaluation findings are in line with earlier evaluation campaigns in NLG: we found weak or no correlations between automatic and human evaluation metrics (Stent et al., 2005; Belz & Reiter, 2006; Novikova et al., 2017a). It is clear that human judgments are essential, but automatic measures have the great benefit that they can be applied quickly and cheaply, and hence are very informative during the develop-

ment of a system. In general, it would be really helpful to have automatic measures that are more indicative of human judgments, which motivates us to pursue new automatic measures to better evaluate automatically generated natural language. One promising option might be the study of referenceless quality estimation models (Dusek et al., 2017). These aim to predict a quality score for a given automatically produced text, instead of comparing it with a gold-standard one, using some automatic metric like BLEU, METEOR or TER.

Recurrent Neural Network Grammars (RNNG) Recently, we have seen the potential of deep neural models in a wide range of NLG-related tasks, like the generation of (first sentences of) Wikipedia entries (e.g., Lebret et al., 2016; Chisholm et al., 2017), poetry (e.g., Zhang & Lapata, 2014), text from abstract meaning representations (e.g., Konstas et al., 2017; Castro Ferreira et al., 2017a) and so on. Although these models have shown an effective performance in these tasks (at least to some extent), they are *a priori* inappropriate models of natural language, since they only sequentially process the linguistic surface, even though words in a sentence can also be organized in terms of nested structures (Dyer et al., 2016). In future work, in order to have more fluent and grammatical texts, we aim to perform NLG using language models like Recurrent Neural Network Grammars (Dyer et al., 2016), which, similar to probabilistic context-free grammars, explicitly take the hierarchical structure among words into account during the generation process.

9.6 Conclusion

In this thesis, we addressed two main challenges for Natural Language Generation systems: how to generate more varied outputs (Chapters 2 - 6),

and how to generate texts from new, more generic input representations (Chapters 7 - 8).

Varied Outputs Based on two new datasets, one for referential forms and one for proper name references, we developed various models that successfully generated varied texts, taking both pragmatic and individual variation into account. In a human evaluation, we found that generating more varied outputs did not have a negative impact on the quality ratings of the texts, although this finding did not generalize to the generation of proper names, where readers seemed to prefer texts in which proper name references followed a similar pattern in similar discourse contexts. To circumvent issues with feature engineering and the integration of different modules, we also introduced an end-to-end REG approach based on deep neural networks, which yielded promising results.

Semantic Inputs Based on two different semantic representations, Abstract Meaning Representations and RDF Triples from the Semantic Web, we developed and evaluated a number of systems that converted these respective semantic input representations into natural language output, comparing both modular and integrated, end-to-end approaches. We conclude that the choice for which representation to use depends on the specific goal of the NLG system under development. AMRs seem better suited for text-to-text NLG or for studying specific issues in the process of generating language, whereas RDF triples may better suit the whole generation process.

Final Remark In this thesis we have studied the automatic generation of more varied output texts, based on various semantic input representations. We hope to have contributed to a better understanding of the NLG process,

paving the way for improved and more engaging automatically generated text.

Summary

Natural Language Generation (NLG) – also known as Automatic Text Generation – is the computational process of generating understandable natural language text from non-linguistic input data (Reiter & Dale, 2000; Gatt & Krahmer, 2018). Practical applications of the process include automatically generated weather forecasts (Goldberg et al., 1994; Sripada et al., 2004; Belz, 2008; Konstas & Lapata, 2013), news written by "robot-journalists" (Clerwall, 2014) and neonatal intensive care reports for doctors and caregivers (Reiter, 2007; Portet et al., 2009).

This thesis focused on two particular problems in the NLG process: how to generate more varied texts to describe the same communicative goal (Chapters 2-6) and what is an appropriate semantic input to generate language from (Chapters 7-8).

For the first problem, we aimed to model linguistic variation in the NLG process, focusing on the generation of noun phrases, a task called Referring Expression Generation (REG). By collecting and analyzing new corpora of referring expressions (described in Chapters 2 and 4), we were able to develop new state-of-the-art data-driven models for two subtasks in modular systems of REG: the choice of referential form (i.e., whether a reference in the text should be a proper name, a pronoun, a description, etc.; Chapter 3) and proper name generation (i.e., given that a reference has the proper name form, should it be the full name of the entity, first name, surname or other proper name form?; Chapter 5). Additionally, we introduced an end-to-end approach, based on neural networks, which, dif-

ferent from modular REG systems, generates varied referring expressions to a discourse entity, deciding on its referential form and content in one shot without explicit feature extraction. Using a new delexicalized version of the WebNLG corpus (Gardent et al., 2017a,b), we showed that the neural model substantially improved over two strong baselines in terms of accuracy of the referring expressions and fluency of the lexicalized texts (Chapter 6).

The second problem addressed in this thesis concerned the input to NLG systems. While there is broad consensus among scholars on the output of NLG systems (i.e., text or speech), there is far less agreement on what the input should be. To address the problem, researchers have started looking for candidate input formats that could be used more broadly within the community. In this thesis, we have looked in detail at two of them: Abstract Meaning Representation (AMR; Chapter 7) and RDF Triples from the semantic web (Chapter 8), which have fundamental differences in terms of level of specification, limitations and availability of resources. To convert both meaning representations into text, we proposed NLG models based on a pipeline architecture as well as models that work in a less modular style, by using methods from Statistical (Koehn et al., 2003) and Neural (Bahdanau et al., 2015) Machine Translation. We concluded that both representations are helpful for NLG research, and which is preferred presumably depends on the specific goal of the NLG system to be developed or on the NLG problem to be addressed. For instance, to study the full textual realization process, working with RDF triples seems preferable over AMRs, while for text-to-text NLG approaches or for the study of specific issues, such as lexical choice or phrase ordering within a sentence, AMRs may be the better choice.

In conclusion, this thesis has focused on the automatic generation of

more varied output texts, based on various semantic input representations. We hope to have contributed to a better understanding of the NLG process, paving the way for improved and more engaging automatically generated text.

Resumo

Geração de Língua Natural (GLN) – também chamada de Geração Automática de Texto – é o processo computacional de geração de língua natural de forma coerente a partir de dados não-linguísticos (Reiter & Dale, 2000; Gatt & Krahmer, 2018). Entre os exemplos de aplicação do processo, encontram-se a geração automática de previsão do tempo (Goldberg et al., 1994; Sripada et al., 2004; Belz, 2008; Konstas & Lapata, 2013), notícias geradas por "robôs jornalistas" (Clerwall, 2014) e relatórios médicos de unidades neonatais intensivas para médicos e enfermeiros (Reiter, 2007; Portet et al., 2009).

Esta tese foca em dois problemas do processo de GLN: o problema de como gerar textos variados para comunicar uma mesma mensagem (Capítulos 2-6) e na escolha de uma representação semântica de entrada apropriada para o processo (Capítulos 7-8).

Para o primeiro problema, nosso objetivo foi modelar a variação linguística no processo de GLN a partir da geração de sintagmas nominais (e.g., expressões de referência), tarefa do processo de GLN conhecida como Geração de Expressões de Referência (GER). A partir da coleta e análise de novos conjuntos de dados de expressões de referência (descritos nos Capítulos 2 e 4), nós desenvolvemos modelos de aprendizado de máquina, e estado da arte, para duas subtarefas do processo de GER: a escolha de formas referenciais (i.e., se uma referência no texto deve assumir a forma de um nome próprio, pronome, descrição, etc.; Capítulo 3) e geração de nomes próprios (i.e., dado que uma referência tem a forma

de um nome próprio, se esta deve ser realizada como o nome completo da entidade em questão, o primeiro nome, o sobrenome, etc.; Capítulo 5). Além disso, nós também introduzimos neste estudo um modelo baseado em redes neurais que, diferente de sistemas modulares de GER, gera variadas expressões de referência para uma entidade no discurso, decidindo sua forma referencial e realização textual de forma conjunta sem a necessidade de extração de features. Usando uma versão delexicalizada no conjunto de dados WebNLG (Gardent et al., 2017a,b), nós mostramos que nosso modelo neural de GER apresenta melhores resultados que dois consideráveis baselines em termos de acurácia das expressões de referência geradas e fluência dos textos lexicalizados pelo modelo (Capítulo 6).

O segundo problema endereçado nesta tese foca na entrada dos sistemas de GLN. Enquanto há um consenso entre pesquisadores com relação à saída destes sistemas (i.e., texto ou áudio), não há um acordo sobre qual é a entrada mais apropriada. Para abordar o problema, alguns pesquisadores têm estudado formatos gerais de entrada, que possam ser usados entre diferentes sistemas. Nesta tese, nós focamos em dois destes: Abstract Meaning Representation (AMR; Capítulo 7) e triplas RDF da web semântica (Capítulo 8). Ambas representações possuem diferenças fundamentais em termos de nível de especificação linguística, limitações e disponibilidade de recursos. Para converter estas duas representações semânticas em texto, nós propusemos sistemas modulares de GLN, assim como modelos baseados em métodos estatísticos (Koehn et al., 2003) e neurais (Bahdanau et al., 2015) de máquina de tradução. Nós concluímos que ambas representações podem ser entradas úteis para sistemas de GLN, e a preferência por uma delas condiciona-se ao escopo do sistema a ser desenvolvido ou ao problema de GLN a ser abordado. Por exemplo, para estudo do processo de realização textual, o uso de triplas RDF é preferível

ao uso das AMRs, enquanto que para aplicações que geram texto a partir de outros textos ou que focam em subtarefas de GLN, como escolha lexical ou ordenação de sintagmas em uma sentença, AMRs são preferíveis.

Em conclusão, esta tese foca na geração automática de textos variados a partir de diferentes representações semânticas. Nós esperamos com este estudo ter contribuído para um melhor entendimento do processo de GLN e para geração automática de textos melhores e mais cativantes para seus leitores.

Acknowledgements

Dr. Sander Wubben and Prof. Dr. Emiel Krahmer for the supervision of this thesis.

The National Council of Scientific and Technological Development from Brazil (CNPq) and the creators of the Science without Borders program for granting this PhD.

Adriana Baltaretu, Ákos Kádár, Diego Moussallem and Iacer Calixto for collaborating with me and co-authoring some of the research projects developed during this period.

Prof. Dr. Ivandré Paraboni, who motivated me to look for and start this journey.

The members of the TiCC language group for reviewing some of my manuscripts and giving insightful comments about them.

My committee, Prof. Dr. Albert Gatt, Prof. Dr. Antal van den Bosch, Dr. Claire Gardent, Dr. Frank Schilder and Dr. Martijn Goudbeek, for their comments in this thesis.

My mother, Angela Maria Castro Ferreira, my sister, Tatiane Castro Ferreira, my father, Tomé Ferreira Neto, my grandmother, Vicentina Festagallo Castro, my aunt, Ivanir Aparecida Ferreira Pinto Spada, for being my family.

My TiCC colleagues for providing a very nice work environment, specially Eva Verschoor and Lauraine de Lima for all the assistance whenever I needed.

My good friends and colleagues who I met in The Netherlands, spe-

cially Katja Helminski (Katjaaaaaaaa*, she will understand when she reads it), Moinuddin M. Haque (Moin), Ayane Santos, René Almeida, Mariana Rachel Dias da Silva (Coração), Mirjam de Haas, Alexandra Sierra, Brenda Szongoth, Evgeny Lavrentjev (Zhenya), Yueqiao Han (Ms. Julie), Adriana Baltaretu, Yevgen Matusevych (Zhenya), Ákos Kádár, Nadine Braun, Priscila Osório Côrtes, Bram Willemsen, Chris Emmery (Metal Chris), Chris van der Lee (Tall Chris), Nanne van Noord, Christine Cook (Chrissy), Laura Capera, Wilma Latuny, Yaser Norouzzadeh Ravari, Yu Gu, Lisa Rombout, Rocsana Bianca and Koel Dutta Chowdhury. Also Fernanda França (Fefa), Jon De Jonge (Jon), Kelly Hessels, Alina Steblovskaya, Dagan Nathaniel Bland and "The master friends" group, which I hope it will become "The friends of the PhD" group: Lívia Riye, Giovana Cremasco, Luis Cristovao, Juliana Almeida, Daniel Guedes, Paula Meira Chinelato (Paulinha), Joana Mattei and Geovana Reis.

My friends from high school and aggregates, Amanda Penna (Mandy), Arthur Grava (Tutu), Bruno Arrabal, Carla Campinas, Caroline Kerestes (Keka), Diogo Gouveia, Flávio Aldana, Gabriele A. de Almeida (Gabs), Juliana Biancheze, Juliana Wes, Kaíssa Nascimento, Mariana Biancheze (Mari), Mariela Ribeiro, Renan Kenji, Renan Rodrigues (Demo), Samira Alvarez Sardella, Vinícius Fernandes (Vini), Vitor Oliveira (Ponto) and the baby members, Rafael Quina Kerestes de Oliveira and Theo Biancheze de Alencar Bravo.

All my friends from Itaú Bank and University of São Paulo (USP), specially Jean Adam Calixto, Márcia Cristina N. Costa, Evaldo Nigro, Marcio Coutinho, Juliana Coutinho Rapanelli, Eder Novais, Mirella Hüne, Alessandro Costa, Rafael Ribeiro, Marcus Viudes, Ricardo Rodrigo Santos (Tetra), Lucas Lima, Luan Kendji, Carlos Eduardo Cagna (Picanha) and João Roisin.

In sum, it was the interactions with all of you, in the many good and bad moments during this period, which resulted in this very special chapter in the book of my life. Thank you!

Publication list

Journal papers

Castro Ferreira, T., & Paraboni, I. (2017). Generating natural language descriptions using speaker-dependent information. *Natural Language Engineering, 23(6)* (pp. 813-834).

Papers in conference proceedings (peer reviewed)

Castro Ferreira, T., Moussallem, D., Kádár, Á., Wubben, S. & Krahmer, E. (2018). NeuralREG: An end-to-end approach to referring expression generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, ACL'2018 (pp. 1959–1969). Melbourne, Australia: Association for Computational Linguistics.

Moussallem, D., Castro Ferreira, T., Zambieri, M., Cavalcanti, M. C., Xexéo, G., Neves, M. & Ngomo, A. N. (2018). RDF2PT: Generating Brazilian Portuguese Texts from RDF Data. In *Proceedings of the 11th edition of the Language Resources and Evaluation Conference*, LREC'18. Miyazaki, Japan.

Castro Ferreira, T., Calixto, I., Wubben, S., & Krahmer, E. (2017). Linguistic realisation as machine translation: Comparing different MT models for AMR-to-text generation. In *Proceedings of the 10th International Conference on Natural Language Generation*, INLG'17 (pp. 1–10). Santiago de Compostela, Spain: Association for Computational Linguistics. **Best Long Paper Award**.

Castro Ferreira, T., & Paraboni, I. (2017). Improving the generation of personalised descriptions. In *Proceedings of the 10th International Conference on Natural Language Generation*, INLG'17 (pp. 233-237). Santiago de Compostela, Spain: Association for

Computational Linguistics.

Castro Ferreira, T., Krahmer, E., & Wubben, S. (2017). Generating flexible proper name references in text: Data, models and evaluation. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, EACL'17 (pp. 655–664). Valencia, Spain: Association for Computational Linguistics.

Baltaretu, A., & Castro Ferreira, T. (2016). Task demands and individual variation in referring expressions. In *Proceedings of the 9th International Natural Language Generation conference*, INLG'16 (pp. 89-93). Edinburgh, Scotland: Association for Computational Linguistics.

Castro Ferreira, T., Wubben, S., & Krahmer, E. (2016). Towards proper name generation: a corpus analysis. In *Proceedings of the 9th International Natural Language Generation conference*, INLG'16 (pp. 222-226). Edinburgh, Scotland: Association for Computational Linguistics.

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*, ACL'16 (pp. 568–577). Berlin, Germany: Association for Computational Linguistics.

Castro Ferreira, T., Krahmer, E., & Wubben, S. (2016). Individual variation in the choice of referential form. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, NAACL-HLT'16 (pp. 423-427). San Diego, California: Association for Computational Linguistics.

Altamirano, R., Ferreira, T., Paraboni, I., & Benotti, L. (2015). Zoom: a corpus of natural language descriptions of map locations. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, ACL-IJCNLP'15 (pp. 69-75). Beijing, China:

Association for Computational Linguistics.

References

Andreas, J. & Klein, D. (2016). Reasoning about pragmatics with neural listeners and speakers. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, EMNLP'16 (pp. 1173–1182). Austin, Texas: Association for Computational Linguistics.

Androutsopoulos, I., Lampouras, G., & Galanis, D. (2013). Generating natural language descriptions from owl ontologies: The natural OWL system. *Journal of Artificial Intelligence Research*, 48(1), 671–715.

Angeli, G., Liang, P., & Klein, D. (2010). A simple domain-independent probabilistic approach to generation. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, EMNLP'10 (pp. 502–512). Cambridge, Massachusetts: Association for Computational Linguistics.

Appelt, D. E. (1980). Problem solving applied to language generation. In *Proceedings of the 18th Annual Meeting on Association for Computational Linguistics*, ACL'80 (pp. 59–63). Philadelphia, Pennsylvania: Association for Computational Linguistics.

Arnold, J. E. (1998). *Reference form and discourse patterns*. PhD thesis, Stanford University Stanford, CA.

Artzi, Y., Lee, K., & Zettlemoyer, L. (2015). Broad-coverage CCG semantic parsing with AMR. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, EMNLP'15 (pp. 1699–1710). Lisbon, Portugal: Association for Computational Linguistics.

Bahdanau, D., Cho, K., & Bengio, Y. (2015). Neural Machine Translation by Jointly Learning to Align and Translate. In *International Conference on Learning Representations*, ICLR'15 San Diego, California.

Banarescu, L., Bonial, C., Cai, S., Georgescu, M., Griffitt, K., Hermjakob, U., Knight, K., Koehn, P., Palmer, M., & Schneider, N. (2013). Abstract meaning representation for

sembanking. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse* (pp. 178–186). Sofia, Bulgaria: Association for Computational Linguistics.

Barzilay, R. & Lapata, M. (2005). Collective content selection for concept-to-text generation. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, HLT'05 (pp. 331–338). Vancouver, British Columbia, Canada: Association for Computational Linguistics.

Barzilay, R. & Lapata, M. (2006). Aggregation via set partitioning for natural language generation. In *Proceedings of the Main Conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, HLT-NAACL'06 (pp. 359–366). New York, New York: Association for Computational Linguistics.

Barzilay, R. & Lee, L. (2004). Catching the drift: Probabilistic content models, with applications to generation and summarization. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, HLT-NAACL'04 (pp. 113–120). Boston, MA, USA: Association for Computational Linguistics.

Bateman, J. A. (1997). Sentence generation and systemic grammar: an introduction. *Iwanami Lecture Series: Language Sciences*, 8, 1–45.

Bateman, J. A. & Paris, C. L. (1989). Phrasing a text in terms the user can understand. In *Proceedings of the 11th International Joint Conference on Artificial Intelligence - Volume 2*, IJCAI'89 (pp. 1511–1517). Detroit, Michigan: Morgan Kaufmann Publishers Inc.

Bayyarapu, H. S. (2011). Efficient algorithm for context sensitive aggregation in natural language generation. In *Proceedings of the International Conference Recent Advances in Natural Language Processing*, RANLP'11 (pp. 84–89). Hissar, Bulgaria: Association for Computational Linguistics.

Belz, A. (2008). Automatic generation of weather forecast texts using comprehensive probabilistic generation-space models. *Natural Language Engineering*, 14(4), 431–455.

Belz, A., Kow, E., Viethen, J., & Gatt, A. (2010). Generating referring expressions in context: The GREC task evaluation challenges. In E. Krahmer & M. Theune (Eds.),

Empirical Methods in Natural Language Generation (pp. 294–327). Berlin, Heidelberg: Springer-Verlag.

Belz, A. & Reiter, E. (2006). Comparing automatic and human evaluation of NLG systems. In *11th Conference of the European Chapter of the Association for Computational Linguistics*, EACL'06 Trento, Italy.

Belz, A., White, M., Espinosa, D., Kow, E., Hogan, D., & Stent, A. (2011). The first surface realisation shared task: Overview and evaluation results. In *Proceedings of the 13th European Workshop on Natural Language Generation* (pp. 217–226). Nancy, France: Association for Computational Linguistics.

Bernardi, R., Cakici, R., Elliott, D., Erdem, A., Erdem, E., Ikizler-Cinbis, N., Keller, F., Muscat, A., & Plank, B. (2016). Automatic Description Generation from Images: A Survey of Models, Datasets, and Evaluation Measures. *Journal of Artificial Intelligence Research*, 55, 409–442.

Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R., & Hellmann, S. (2009). DBpedia - a crystallization point for the web of data. *Web Semantics: Science, Services and Agents on the World Wide Web*, 7(3), 154 – 165. The Web of Data.

Bohnet, B. (2008). The fingerprint of human referring expressions and their surface realization with graph transducers. In *Proceedings of the Fifth International Natural Language Generation Conference*, INLG'08 (pp. 207–210). Salt Fork, Ohio: Association for Computational Linguistics.

Bojar, O., Chatterjee, R., Federmann, C., Haddow, B., Huck, M., Hokamp, C., Koehn, P., Logacheva, V., Monz, C., Negri, M., Post, M., Scarton, C., Specia, L., & Turchi, M. (2015). Findings of the 2015 Workshop on Statistical Machine Translation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation* (pp. 1–46). Lisbon, Portugal: Association for Computational Linguistics.

Bos, J. (2016). Expressive power of abstract meaning representations. *Computational Linguistics*, 42(3), 527–535.

Brennan, S. E. (1995). Centering attention in discourse. *Language and Cognitive Processes*, 10(2), 137–167.

Callaway, C. B. & Lester, J. C. (2002). Pronominalization in generated discourse and dialogue. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL'02 (pp. 88–95). Philadelphia, Pennsylvania: Association for Computational Linguistics.

Castro Ferreira, T., Calixto, I., Wubben, S., & Krahmer, E. (2017a). Linguistic realisation as machine translation: Comparing different MT models for AMR-to-text generation. In *Proceedings of the 10th International Conference on Natural Language Generation*, INLG'17 (pp. 1–10). Santiago de Compostela, Spain: Association for Computational Linguistics.

Castro Ferreira, T., Krahmer, E., & Wubben, S. (2016a). Individual variation in the choice of referential form. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, HLT-NAACL'16 (pp. 423–427). San Diego, California: Association for Computational Linguistics.

Castro Ferreira, T., Krahmer, E., & Wubben, S. (2016b). 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*, ACL'16 (pp. 568—577). Berlin, Germany: Association for Computational Linguistics.

Castro Ferreira, T., Krahmer, E., & Wubben, S. (2017b). Generating flexible proper name references in text: Data, models and evaluation. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, EACL'17 (pp. 655–664). Valencia, Spain: Association for Computational Linguistics.

Castro Ferreira, T. & Paraboni, I. (2014). Referring expression generation: Taking speakers' preferences into account. In P. Sojka, A. Horák, I. KopeČek, & K. Pala (Eds.), *Text, Speech and Dialogue*, volume 8655 of *Lecture Notes in Computer Science* (pp. 539–546). Springer International Publishing.

Castro Ferreira, T., Wubben, S., & Krahmer, E. (2016c). Towards proper name generation: a corpus analysis. In *Proceedings of the 9th International Natural Language Generation conference*, INLG'16 (pp. 222–226). Edinburgh, UK: Association for Computational Linguistics.

Chafe, W. (1994). *Discourse, Consciousness, and Time: The Flow and Displacement of Conscious Experience in Speaking and Writing*. University of Chicago Press.

Chen, D. L. & Mooney, R. J. (2008). Learning to sportscast: A test of grounded language acquisition. In *Proceedings of the 25th International Conference on Machine Learning*, ICML'08 (pp. 128–135). Helsinki, Finland: ACM.

Cherry, C. & Foster, G. (2012). Batch tuning strategies for statistical machine translation. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, NAACL-HLT'12 (pp. 427–436). Montreal, Canada: Association for Computational Linguistics.

Chisholm, A., Radford, W., & Hachey, B. (2017). Learning to generate one-sentence biographies from wikidata. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, EACL'17 (pp. 633–642). Valencia, Spain: Association for Computational Linguistics.

Cho, K., van Merrienboer, B., Bahdanau, D., & Bengio, Y. (2014). On the Properties of Neural Machine Translation: Encoder–Decoder Approaches. In *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation* (pp. 103–111). Doha, Qatar.

Clark, J. H., Dyer, C., Lavie, A., & Smith, N. A. (2011). Better Hypothesis Testing for Statistical Machine Translation: Controlling for Optimizer Instability. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers - Volume 2*, ACL'11 (pp. 176–181). Portland, Oregon.

Clerwall, C. (2014). Enter the robot journalist. *Journalism Practice*, 8(5), 519–531.

Dahl, D. A., Bates, M., Brown, M., Fisher, W., Hunicke-Smith, K., Pallett, D., Pao, C., Rudnicky, A., & Shriberg, E. (1994). Expanding the scope of the ATIS task: The ATIS-3 corpus. In *Proceedings of the Workshop on Human Language Technology*, HLT'94 (pp. 43–48). Plainsboro, NJ: Association for Computational Linguistics.

Dale, R. & Haddock, N. (1991). Generating referring expressions involving relations. In *Proceedings of the fifth conference on European chapter of the Association for Computational Linguistics*, EACL'91 (pp. 161–166). Berlin, Germany: Association for Computational Linguistics.

Dale, R. & Reiter, E. (1995). Computational interpretations of the gricean maxims in the generation of referring expressions. *Cognitive science*, 19(2), 233–263.

Dale, R. & Viethen, J. (2010). Attribute-centric referring expression generation. In *Empirical methods in natural language generation* (pp. 163–179). Springer.

Dong, L., Huang, S., Wei, F., Lapata, M., Zhou, M., & Xu, K. (2017). Learning to generate product reviews from attributes. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, EACL'17 (pp. 623–632). Valencia, Spain: Association for Computational Linguistics.

Duma, D. & Klein, E. (2013). Generating natural language from linked data: Unsupervised template extraction. In *Proceedings of the 10th International Conference on Computational Semantics (IWCS 2013) – Long Papers* (pp. 83–94). Potsdam, Germany: Association for Computational Linguistics.

Dusek, O., Novikova, J., & Rieser, V. (2017). Referenceless quality estimation for natural language generation. *CoRR*, abs/1708.01759.

Dyer, C., Kuncoro, A., Ballesteros, M., & Smith, N. A. (2016). Recurrent neural network grammars. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, HLT-NAACL'16 (pp. 199–209). San Diego, California: Association for Computational Linguistics.

Ferres, L., Parush, A., Roberts, S., & Lindgaard, G. (2006). Helping people with visual impairments gain access to graphical information through natural language: The igraph system. In *Proceedings of the 10th International Conference on Computers Helping People with Special Needs*, ICCHP'06 (pp. 1122–1130). Linz, Austria: Springer-Verlag.

Ficler, J. & Goldberg, Y. (2017). Controlling linguistic style aspects in neural language generation. In *Proceedings of the Workshop on Stylistic Variation* (pp. 94–104). Copenhagen, Denmark: Association for Computational Linguistics.

Finkel, J. R., Grenager, T., & Manning, C. (2005). Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*, ACL'05 (pp. 363–370). Ann Arbor, Michigan: Association for Computational Linguistics.

Flanigan, J., Dyer, C., Smith, N. A., & Carbonell, J. (2016). Generation from abstract meaning representation using tree transducers. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, HLT-NAACL'16 (pp. 731–739). San Diego, California: Association for Computational Linguistics.

Flanigan, J., Thomson, S., Carbonell, J., Dyer, C., & Smith, N. A. (2014). A discriminative graph-based parser for the abstract meaning representation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, ACL'14 (pp. 1426–1436). Baltimore, Maryland: Association for Computational Linguistics.

Friedman, J. (1969). Directed random generation of sentences. *Commun. ACM*, 12(1), 40–46.

Gal, Y. & Ghahramani, Z. (2016). A Theoretically Grounded Application of Dropout in Recurrent Neural Networks. In *Advances in Neural Information Processing Systems, NIPS* (pp. 1019–1027). Barcelona, Spain.

Galley, M. & Manning, C. D. (2008). A simple and effective hierarchical phrase reordering model. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, EMNLP'08 (pp. 848–856). Honolulu, Hawaii: Association for Computational Linguistics.

Gao, Q. & Vogel, S. (2008). Parallel implementations of word alignment tool. In *Software Engineering, Testing, and Quality Assurance for Natural Language Processing*, SETQA-NLP'08 (pp. 49–57). Columbus, Ohio: Association for Computational Linguistics.

Gardent, C., Shimorina, A., Narayan, S., & Perez-Beltrachini, L. (2017a). Creating training corpora for NLG micro-planners. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, ACL'17 (pp. 179–188). Vancouver, Canada: Association for Computational Linguistics.

Gardent, C., Shimorina, A., Narayan, S., & Perez-Beltrachini, L. (2017b). The WebNLG challenge: Generating text from RDF data. In *Proceedings of the 10th International Conference on Natural Language Generation*, INLG'17 (pp. 124–133). Santiago de Compostela, Spain: Association for Computational Linguistics.

Gatt, A. & Belz, A. (2010). Introducing shared tasks to NLG: The TUNA shared task evaluation challenges. In E. Krahmer & M. Theune (Eds.), *Empirical Methods in Natural Language Generation* (pp. 264–293). Berlin, Heidelberg: Springer-Verlag.

Gatt, A. & Krahmer, E. (2018). Survey of the state of the art in natural language generation: Core tasks, applications and evaluation. *Journal of Artificial Intelligence Research*, 61, 65–170.

Gkatzia, D., Hastie, H., & Lemon, O. (2014). Comparing multi-label classification with reinforcement learning for summarisation of time-series data. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, ACL'14 (pp. 1231–1240). Baltimore, Maryland: Association for Computational Linguistics.

Glorot, X. & Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, volume 9 of *Proceedings of Machine Learning Research* (pp. 249–256). Chia Laguna Resort, Sardinia, Italy: PMLR.

Goldberg, E., Driedger, N., & Kittredge, R. I. (1994). Using natural-language processing to produce weather forecasts. *IEEE Expert: Intelligent Systems and Their Applications*, 9(2), 45–53.

Greenbacker, C. F. & McCoy, K. F. (2009). Feature selection for reference generation as informed by psycholinguistic research. In *Proceedings of the CogSci 2009 Workshop on Production of Referring Expressions*, PRE-Cogsci'09.

Greenbacker, C. F., Sparks, N. L., McCoy, K. F., & Kuo, C.-Y. (2010). Udel: Refining a method of named entity generation. In *Proceedings of the 6th International Natural Language Generation Conference*, INLG'10 (pp. 239–240). Trim, Co. Meath, Ireland: Association for Computational Linguistics.

Grosz, B. J., Weinstein, S., & Joshi, A. K. (1995). Centering: A framework for modeling the local coherence of discourse. *Computational Linguistics*, 21(2), 203–225.

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

Gupta, S. & Bandopadhyay, S. (2009). JUNLG-MSR: A machine learning approach of main subject reference selection with rule based improvement. In *Proceedings of the*

2009 Workshop on Language Generation and Summarisation, UCNLG+Sum'09 (pp. 103–104). Suntec, Singapore: Association for Computational Linguistics.

Gyawali, B. & Gardent, C. (2014). Surface realisation from knowledge-bases. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, ACL'14 (pp. 424–434). Baltimore, Maryland: Association for Computational Linguistics.

Heafield, K., Pouzyrevsky, I., Clark, J. H., & Koehn, P. (2013). Scalable modified kneser-ney language model estimation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, ACL'13 (pp. 690–696). Sofia, Bulgaria: Association for Computational Linguistics.

Henschel, R., Cheng, H., & Poesio, M. (2000). Pronominalization revisited. In *Proceedings of the 18th Conference on Computational Linguistics - Volume 1*, COLING'00 (pp. 306–312). Saarbrücken, Germany: Association for Computational Linguistics.

Hochreiter, S. & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8), 1735–1780.

Hovy, E. H. (1990). Pragmatics and natural language generation. *Artificial Intelligence*, 43(2), 153–197.

Iordanskaja, L., Kim, M., Kittredge, R., Lavoie, B., & Polguère, A. (1992). Generation of extended bilingual statistical reports. In *Proceedings of the 14th Conference on Computational Linguistics - Volume 3*, COLING'92 (pp. 1019–1023). Nantes, France: Association for Computational Linguistics.

Jaeger, T. F. (2010). Redundancy and reduction: Speakers manage syntactic information density. *Cognitive psychology*, 61(1), 23–62.

Kate, R. J., Wong, Y. W., & Mooney, R. J. (2005). Learning to transform natural to formal languages. In *Proceedings of the 20th National Conference on Artificial Intelligence - Volume 3*, AAAI'05 (pp. 1062–1068). Pittsburgh, Pennsylvania: AAAI Press.

Kim, J. & Mooney, R. J. (2010). Generative alignment and semantic parsing for learning from ambiguous supervision. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, COLING'10 (pp. 543–551). Beijing, China: Association for Computational Linguistics.

Koehn, P., Axelrod, A., Birch, A., Callison-Burch, C., Osborne, M., & Talbot, D. (2005). Edinburgh System Description for the 2005 IWSLT Speech Translation Evaluation. In *International Workshop on Spoken Language Translation*.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., & Herbst, E. (2007). Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, ACL'07 (pp. 177–180). Prague, Czech Republic: Association for Computational Linguistics.

Koehn, P. & Monz, C. (2006). Manual and automatic evaluation of machine translation between european languages. In *Proceedings on the Workshop on Statistical Machine Translation* (pp. 102–121). New York City: Association for Computational Linguistics.

Koehn, P., Och, F. J., & Marcu, D. (2003). Statistical phrase-based translation. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, NAACL'03 (pp. 48–54). Edmonton, Canada: Association for Computational Linguistics.

Koller, A. & Striegnitz, K. (2002). Generation as dependency parsing. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL'02 (pp. 17–24). Philadelphia, Pennsylvania: Association for Computational Linguistics.

Konstantinova, N., de Sousa, S. C. M., Díaz, N. P. C., López, M. J. M., Taboada, M., & Mitkov, R. (2012). A review corpus annotated for negation, speculation and their scope. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation*, LREC'12 (pp. 3190–3195). Istanbul, Turkey.

Konstas, I., Iyer, S., Yatskar, M., Choi, Y., & Zettlemoyer, L. (2017). Neural AMR: Sequence-to-sequence models for parsing and generation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, ACL'17 (pp. 146–157). Vancouver, Canada: Association for Computational Linguistics.

Konstas, I. & Lapata, M. (2013). A global model for concept-to-text generation. *Journal of Artificial Intelligence Research*, 48(1), 305–346.

Krahmer, E. & Theune, M. (2002). Efficient context-sensitive generation of referring expressions. In K. van Deemter & R. Kibble (Eds.), *Information sharing: Reference*

and presupposition in language generation and interpretation (pp. 223–264). Stanford, CA: CSLI.

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

Krahmer, E., van Erk, S., & Verleg, A. (2003). Graph-based generation of referring expressions. *Computational Linguistics*, 29(1), 53–72.

Kukich, K. (1983). Design of a knowledge-based report generator. In *Proceedings of the 21st Annual Meeting on Association for Computational Linguistics*, ACL'83 (pp. 145–150). Cambridge, Massachusetts: Association for Computational Linguistics.

Kullback, S. (1997). *Information Theory and Statistics*. A Wiley publication in mathematical statistics. Dover Publications.

Lavie, A. & Agarwal, A. (2007). Meteor: An automatic metric for MT evaluation with high levels of correlation with human judgments. In *Proceedings of the Second Workshop on Statistical Machine Translation*, StatMT'07 (pp. 228–231). Prague, Czech Republic.

Lebret, R., Grangier, D., & Auli, M. (2016). Neural text generation from structured data with application to the biography domain. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, EMNLP'16 (pp. 1203–1213). Austin, Texas: Association for Computational Linguistics.

Lerner, U. & Petrov, S. (2013). Source-side classifier preordering for machine translation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, EMNLP'13 (pp. 513–523). Seattle, Washington, USA: Association for Computational Linguistics.

Levenshtein, V. I. (1966). Binary Codes Capable of Correcting Deletions, Insertions and Reversals. *Soviet Physics Doklady*, 10, 707.

Liang, P., Jordan, M. I., & Klein, D. (2009). Learning semantic correspondences with less supervision. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1 - Volume 1*, ACL'09 (pp. 91–99). Suntec, Singapore: Association for Computational Linguistics.

Libovický, J. & Helcl, J. (2017). Attention strategies for multi-source sequence-to-sequence learning. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, ACL'17 (pp. 196–202). Vancouver, Canada: Association for Computational Linguistics.

Lin, J. (1991). Divergence measures based on the shannon entropy. *Information Theory, IEEE Transactions on*, 37(1), 145–151.

Lu, W., Ng, H. T., & Lee, W. S. (2009). Natural language generation with tree conditional random fields. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1 - Volume 1*, EMNLP'09 (pp. 400–409). Singapore: Association for Computational Linguistics.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., & McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations* (pp. 55–60).

Marcus, M., Kim, G., Marcinkiewicz, M. A., MacIntyre, R., Bies, A., Ferguson, M., Katz, K., & Schasberger, B. (1994). The Penn treebank: Annotating predicate argument structure. In *Proceedings of the Workshop on Human Language Technology*, HLT'94 (pp. 114–119). Plainsboro, NJ: Association for Computational Linguistics.

McKeown, K. R. (1982). The TEXT system for natural language generation: An overview. In *Proceedings of the 20th Annual Meeting on Association for Computational Linguistics*, ACL'82 (pp. 113–120). Toronto, Ontario, Canada: Association for Computational Linguistics.

Mei, H., Bansal, M., & Walter, M. R. (2016). What to talk about and how? selective generation using LSTMs with coarse-to-fine alignment. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, HLT-NAACL'16 (pp. 720–730). San Diego, California: Association for Computational Linguistics.

Mesnil, G., He, X., Deng, L., & Bengio, Y. (2013). Investigation of recurrent-neural-network architectures and learning methods for spoken language understanding. In *14th Annual Conference of the International Speech Communication Association*, INTERSPEECH'13 (pp. 3771–3775). Lyon, France.

Moussallem, D., Usbeck, R., Röder, M., & Ngonga Ngomo, A.-C. (2017). MAG: A Multilingual, Knowledge-base Agnostic and Deterministic Entity Linking Approach. *ArXiv e-prints*.

Neubig, G., Dyer, C., Goldberg, Y., Matthews, A., Ammar, W., Anastasopoulos, A., Ballesteros, M., Chiang, D., Clothiaux, D., Cohn, T., Duh, K., Faruqui, M., Gan, C., Garrette, D., Ji, Y., Kong, L., Kuncoro, A., Kumar, G., Malaviya, C., Michel, P., Oda, Y., Richardson, M., Saphra, N., Swayamdipta, S., & Yin, P. (2017). DyNet: The Dynamic Neural Network Toolkit. *ArXiv e-prints*.

Novikova, J., Dušek, O., Cercas Curry, A., & Rieser, V. (2017a). Why we need new evaluation metrics for NLG. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, EMNLP'17 (pp. 2231–2242). Copenhagen, Denmark: Association for Computational Linguistics.

Novikova, J., Dusek, O., & Rieser, V. (2017b). Data-driven natural language generation: Paving the road to success. *CoRR*, abs/1706.09433.

Novikova, J., Dusek, O., & Rieser, V. (2017c). The E2E dataset: New challenges for end-to-end generation. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue* (pp. 201–206). Saarbrücken, Germany.

Och, F. J. (2003). Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, ACL'03 (pp. 160–167). Sapporo, Japan: Association for Computational Linguistics.

Okazaki, N. (2007). CRFsuite: a fast implementation of conditional random fields (CRFs).

Orita, N., Vornov, E., Feldman, N., & Daumé III, H. (2015). Why discourse affects speakers' choice of referring expressions. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, ACL'15 (pp. 1639–1649). Beijing, China: Association for Computational Linguistics.

Orăsan, C. & Dornescu, I. (2009). WLV: A confidence-based machine learning method for the GREC-NEG'09 task. In *Proceedings of the 2009 Workshop on Language Generation and Summarisation*, UCNLG+Sum'09 (pp. 107–108). Suntec, Singapore: Association for Computational Linguistics.

Papineni, K., Roukos, S., Ward, T., & Zhu, W.-J. (2002). Bleu: a method for automatic evaluation of machine translation. In *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*, ACL'02 (pp. 311–318). Philadelphia, Pennsylvania, USA: Association for Computational Linguistics.

Portet, F., Reiter, E., Gatt, A., Hunter, J., Sripada, S., Freer, Y., & Sykes, C. (2009). Automatic generation of textual summaries from neonatal intensive care data. *Artificial Intelligence*, 173(7–8), 789 – 816.

Pourdamghani, N., Gao, Y., Hermjakob, U., & Knight, K. (2014). Aligning english strings with abstract meaning representation graphs. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, EMNLP'14 (pp. 425–429). Doha, Qatar: Association for Computational Linguistics.

Pourdamghani, N., Knight, K., & Hermjakob, U. (2016). Generating english from abstract meaning representations. In *Proceedings of the 9th International Natural Language Generation conference*, INLG'16 (pp. 21–25). Edinburgh, UK: Association for Computational Linguistics.

Reiter, E. (2007). An architecture for data-to-text systems. In *Proceedings of the Eleventh European Workshop on Natural Language Generation*, ENLG'07 (pp. 97–104). Germany: Association for Computational Linguistics.

Reiter, E. & Dale, R. (2000). *Building natural language generation systems*. New York, NY, USA: Cambridge University Press.

Reiter, E., Robertson, R., & Osman, L. M. (2003). Lessons from a failure: Generating tailored smoking cessation letters. *Artificial Intelligence*, 144(1), 41 – 58.

Reiter, E., Sripada, S., Hunter, J., Yu, J., & Davy, I. (2005). Choosing words in computer-generated weather forecasts. *Artificial Intelligence*, 167(1-2), 137–169.

Schuster, M. & Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11), 2673–2681.

Sennrich, R., Haddow, B., & Birch, A. (2016a). Edinburgh neural machine translation systems for wmt 16. In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers* (pp. 371–376). Berlin, Germany: Association for Computational Linguistics.

Sennrich, R., Haddow, B., & Birch, A. (2016b). Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, ACL'16 (pp. 1715–1725). Berlin, Germany: Association for Computational Linguistics.

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

Simmons, R. & Slocum, J. (1972). Generating english discourse from semantic networks. *Commun. ACM*, 15(10), 891–905.

Singh, S., Subramanya, A., Pereira, F., & McCallum, A. (2012). *Wikilinks: A Large-scale Cross-Document Coreference Corpus Labeled via Links to Wikipedia*. Technical Report UM-CS-2012-015.

Smiley, C., Plachouras, V., Schilder, F., Bretz, H., Leidner, J., & Song, D. (2016). When to plummet and when to soar: Corpus based verb selection for natural language generation. In *Proceedings of the 9th International Natural Language Generation conference*, INLG'16 (pp. 36–39). Edinburgh, UK: Association for Computational Linguistics.

Snover, M., Dorr, B., Schwartz, R., Micciulla, L., & Makhoul, J. (2006). A study of translation edit rate with targeted human annotation. In *Proceedings of Association for Machine Translation in the Americas, AMTA* (pp. 223–231). Cambridge, MA, USA.

Song, L., Peng, X., Zhang, Y., Wang, Z., & Gildea, D. (2017). AMR-to-text generation with synchronous node replacement grammar. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, ACL'17 (pp. 7–13). Vancouver, Canada: Association for Computational Linguistics.

Song, L., Zhang, Y., Peng, X., Wang, Z., & Gildea, D. (2016). AMR-to-text generation as a traveling salesman problem. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, EMNLP'16 (pp. 2084–2089). Austin, Texas: Association for Computational Linguistics.

Specia, L., Frank, S., Sima'an, K., & Elliott, D. (2016). A shared task on multimodal machine translation and crosslingual image description. In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers* (pp. 543–553). Berlin, Germany: Association for Computational Linguistics.

Sripada, S. & Gao, F. (2007). Summarizing dive computer data: A case study in integrating textual and graphical presentations of numerical data. In *Workshop on Multimodal Output Generation*, MOG'07 (pp. 149–157).: Association for Computational Linguistics.

Sripada, S., Reiter, E., & Davy, I. (2004). SumTime-Mousam: Configurable marine weather forecast generator.

Stent, A., Marge, M., & Singhai, M. (2005). Evaluating evaluation methods for generation in the presence of variation. In *Proceedings of the 6th International Conference on Computational Linguistics and Intelligent Text Processing*, CICLing'05 (pp. 341–351). Mexico City, Mexico: Springer-Verlag.

Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems* (pp. 3104–3112). Montreal, Quebec, Canada.

Theune, M., Klabbers, E., De Pijper, J. R., Krahmer, E., & Odijk, J. (2001). From data to speech: a general approach. *Natural Language Engineering*, 7(1), 47–86.

Turner, R., Sripada, S., Reiter, E., & Davy, I. P. (2006). Generating spatio-temporal descriptions in pollen forecasts. In *Proceedings of the Eleventh Conference of the European Chapter of the Association for Computational Linguistics: Demonstrations*, EACL'06 (pp. 163–166). Trento, Italy: Association for Computational Linguistics.

van Deemter, K. (2014). Referability. In A. Stent & S. Bangalore (Eds.), *Natural Language Generation in Interactive Systems* chapter 5, (pp. 101–103). New York, NY, USA: Cambridge University Press.

van Deemter, K. (2016). Designing algorithms for referring with proper names. In *Proceedings of the 9th International Natural Language Generation conference*, INLG'16 (pp. 31–35). Edinburgh, UK: Association for Computational Linguistics.

Van Deemter, K., Gatt, A., van Gompel, R. P., & Krahmer, E. (2012). Toward a computational psycholinguistics of reference production. *Topics in cognitive science*, 4(2), 166–183.

van der Lee, C., Krahmer, E., & Wubben, S. (2017). Pass: A dutch data-to-text system for soccer, targeted towards specific audiences. In *Proceedings of the 10th International*

Conference on Natural Language Generation, INLG'2017 (pp. 95–104). Santiago de Compostela, Spain: Association for Computational Linguistics.

Viethen, J. & Dale, R. (2010). Speaker-dependent variation in content selection for referring expression generation. In *Proceedings of the Australasian Language Technology Association Workshop 2010* (pp. 81–89). Melbourne, Australia.

Vrandečić, D. & Krötzsch, M. (2014). Wikidata: A free collaborative knowledgebase. *Commun. ACM*, 57(10), 78–85.

Štajner, S., Calixto, I., & Saggion, H. (2015). Automatic text simplification for spanish: Comparative evaluation of various simplification strategies. In *Proceedings of the International Conference Recent Advances in Natural Language Processing* (pp. 618–626). Hissar, Bulgaria.

Wen, T.-H., Gašić, M., Mrkšić, N., Rojas-Barahona, L. M., Su, P.-H., Vandyke, D., & Young, S. (2016). Multi-domain neural network language generation for spoken dialogue systems. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, HLT-NAACL'16 (pp. 120–129). San Diego, California: Association for Computational Linguistics.

Wen, T.-H., Gasic, M., Mrkšić, N., Su, P.-H., Vandyke, D., & Young, S. (2015). Semantically conditioned LSTM-based natural language generation for spoken dialogue systems. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, EMNLP'15 (pp. 1711–1721). Lisbon, Portugal: Association for Computational Linguistics.

White, M. & Rajkumar, R. (2008). A more precise analysis of punctuation for broad-coverage surface realization with CCG. In *Coling 2008: Proceedings of the workshop on Grammar Engineering Across Frameworks* (pp. 17–24). Manchester, England: Coling 2008 Organizing Committee.

Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., Klingner, J., Shah, A., Johnson, M., Liu, X., Łukasz Kaiser, Gouws, S., Kato, Y., Kudo, T., Kazawa, H., Stevens, K., Kurian, G., Patil, N., Wang, W., Young, C., Smith, J., Riesa, J., Rudnick, A., Vinyals, O., Corrado, G., Hughes, M., & Dean, J. (2016). Google's neural machine translation system: Bridging the gap between human and machine translation. *CoRR*, abs/1609.08144.

Wubben, S., van den Bosch, A., & Krahmer, E. (2012). Sentence simplification by monolingual machine translation. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers - Volume 1*, ACL'12 (pp. 1015–1024). Jeju Island, Korea: Association for Computational Linguistics.

Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., Zemel, R., & Bengio, Y. (2015). Show, attend and tell: Neural image caption generation with visual attention. In D. Blei & F. Bach (Eds.), *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)* (pp. 2048–2057).: JMLR Workshop and Conference Proceedings.

Yeh, C.-L. & Mellish, C. (1997). An empirical study on the generation of anaphora in chinese. *Computational Linguistics*, 23(1), 171–190.

Yngve, V. H. (1961). *Random generation of English sentences*. Massachusetts Inst. of Technology.

Yu, J., Reiter, E., Hunter, J., & Mellish, C. (2007). Choosing the content of textual summaries of large time-series data sets. *Natural Language Engineering*, 13(1), 25–49.

Zarriess, S. & Kuhn, J. (2013). Combining Referring Expression Generation and Surface Realization: A Corpus-Based Investigation of Architectures. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, ACL'13 (pp. 1547–1557). Sofia, Bulgaria: Association for Computational Linguistics.

Zeiler, M. D. (2012). ADADELTA: An adaptive learning rate method. *CoRR*, abs/1212.5701.

Zhang, X. & Lapata, M. (2014). Chinese poetry generation with recurrent neural networks. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, EMNLP'14 (pp. 670–680). Doha, Qatar: Association for Computational Linguistics.

TiCC PhD Series

1. Pashiera Barkhuysen. Audiovisual Prosody in Interaction. Promotores: M.G.J. Swerts, E.J. Krahmer. Tilburg, 3 October 2008.
2. Ben Torben-Nielsen. Dendritic Morphology: Function Shapes Structure. Promotores: H.J. van den Herik, E.O. Postma. Co-promotor: K.P. Tuyls. Tilburg, 3 December 2008.
3. Hans Stol. A Framework for Evidence-based Policy Making Using IT. Promotor: H.J. van den Herik. Tilburg, 21 January 2009.
4. Jeroen Geertzen. Dialogue Act Recognition and Prediction. Promotor: H. Bunt. Co-promotor: J.M.B. Terken. Tilburg, 11 February 2009.
5. Sander Canisius. Structured Prediction for Natural Language Processing. Promotores: A.P.J. van den Bosch, W. Daelemans. Tilburg, 13 February 2009.
6. Fritz Reul. New Architectures in Computer Chess. Promotor: H.J. van den Herik. Co-promotor: J.W.H.M. Uiterwijk. Tilburg, 17 June 2009.
7. Laurens van der Maaten. Feature Extraction from Visual Data. Promotores: E.O. Postma, H.J. van den Herik. Co-promotor: A.G. Lange. Tilburg, 23 June 2009 (cum laude).
8. Stephan Raaijmakers. Multinomial Language Learning. Promotores: W. Daelemans, A.P.J. van den Bosch. Tilburg, 1 December 2009.
9. Igor Berezhnoy. Digital Analysis of Paintings. Promotores: E.O. Postma, H.J. van den Herik. Tilburg, 7 December 2009.
10. Toine Bogers. Recommender Systems for Social Bookmarking. Promotor: A.P.J. van den Bosch. Tilburg, 8 December 2009.
11. Sander Bakkes. Rapid Adaptation of Video Game AI. Promotor: H.J. van den Herik. Co- promotor: P. Spronck. Tilburg, 3 March 2010.
12. Maria Mos. Complex Lexical Items. Promotor: A.P.J. van den Bosch. Co-promotores: A. Vermeer, A. Backus. Tilburg, 12 May 2010 (in collaboration

with the Department of Language and Culture Studies).

13. Marieke van Erp. Accessing Natural History. Discoveries in data cleaning, structuring, and retrieval. Promotor: A.P.J. van den Bosch. Co-promotor: P.K. Lendvai. Tilburg, 30 June 2010.
14. Edwin Commandeur. Implicit Causality and Implicit Consequentiality in Language Comprehension. Promotores: L.G.M. Noordman, W. Vonk. Co-promotor: R. Cozijn. Tilburg, 30 June 2010.
15. Bart Bogaert. Cloud Content Contention. Promotores: H.J. van den Herik, E.O. Postma. Tilburg, 30 March 2011.
16. Xiaoyu Mao. Airport under Control. Promotores: H.J. van den Herik, E.O. Postma. Co- promotores: N. Roos, A. Salden. Tilburg, 25 May 2011.
17. Olga Petukhova. Multidimensional Dialogue Modelling. Promotor: H. Bunt. Tilburg, 1 September 2011.
18. Lisette Mol. Language in the Hands. Promotores: E.J. Krahmer, A.A. Maes, M.G.J. Swerts. Tilburg, 7 November 2011 (cum laude).
19. Herman Stehouwer. Statistical Language Models for Alternative Sequence Selection. Promotores: A.P.J. van den Bosch, H.J. van den Herik. Co-promotor: M.M. van Zaanen. Tilburg, 7 December 2011.
20. Terry Kakeeto-Aelen. Relationship Marketing for SMEs in Uganda. Promotores: J. Chr. van Dalen, H.J. van den Herik. Co-promotor: B.A. Van de Walle. Tilburg, 1 February 2012.
21. Suleman Shahid. Fun & Face: Exploring non-verbal expressions of emotion during playful interactions. Promotores: E.J. Krahmer, M.G.J. Swerts. Tilburg, 25 May 2012.
22. Thijs Vis. Intelligence, Politie en Veiligheidsdienst: Verenigbare Grootheden? Promotores: T.A. de Roos, H.J. van den Herik, A.C.M. Spapens. Tilburg, 6 June 2012 (in collaboration with the Tilburg School of Law).
23. Nancy Pascall. Engendering Technology Empowering Women. Promotores: H.J. van den Herik, M. Diocaretz. Tilburg, 19 November 2012.
24. Agus Gunawan. Information Access for SMEs in Indonesia. Promotor: H.J. van den Herik. Co- promotores: M. Wahdan, B.A. Van de Walle. Tilburg, 19 December 2012.

25. Giel van Lankveld. Quantifying Individual Player Differences. Promotores: H.J. van den Herik, A.R. Arntz. Co-promotor: P. Spronck. Tilburg, 27 February 2013.
26. Sander Wubben. Text-to-text Generation Using Monolingual Machine Translation. Promotores: E.J. Krahmer, A.P.J. van den Bosch, H. Bunt. Tilburg, 5 June 2013.
27. Jeroen Janssens. Outlier Selection and One-Class Classification. Promotores: E.O. Postma, H.J. van den Herik. Tilburg, 11 June 2013.
28. Martijn Balsters. Expression and Perception of Emotions: The Case of Depression, Sadness and Fear. Promotores: E.J. Krahmer, M.G.J. Swerts, A.J.J.M. Vingerhoets. Tilburg, 25 June 2013.
29. Lisanne van Weelden. Metaphor in Good Shape. Promotor: A.A. Maes. Co-promotor: J. Schilperoord. Tilburg, 28 June 2013.
30. Ruud Koolen. "Need I say More? On Overspecification in Definite Reference." Promotores: E.J. Krahmer, M.G.J. Swerts. Tilburg, 20 September 2013.
31. J. Douglas Mastin. Exploring Infant Engagement. Language Socialization and Vocabulary. Development: A Study of Rural and Urban Communities in Mozambique. Promotor: A.A. Maes. Co-promotor: P.A. Vogt. Tilburg, 11 October 2013.
32. Philip C. Jackson. Jr. Toward Human-Level Artificial Intelligence – Representation and Computation of Meaning in Natural Language. Promotores: H.C. Bunt, W.P.M. Daelemans. Tilburg, 22 April 2014.
33. Jorrig Vogels. Referential choices in language production: The Role of Accessibility. Promotores: A.A. Maes, E.J. Krahmer. Tilburg, 23 April 2014.
34. Peter de Kock. Anticipating Criminal Behaviour. Promotores: H.J. van den Herik, J.C. Scholtes. Co-promotor: P. Spronck. Tilburg, 10 September 2014.
35. Constantijn Kaland. Prosodic marking of semantic contrasts: do speakers adapt to addressees? Promotores: M.G.J. Swerts, E.J. Krahmer. Tilburg, 1 October 2014.
36. Jasmina Marić. Web Communities, Immigration and Social Capital. Promotor: H.J. van den Herik. Co-promotores: R. Cozijn, M. Spotti. Tilburg, 18 November 2014.
37. Pauline Meesters. Intelligent Blauw. Promotores: H.J. van den Herik, T.A. de Roos. Tilburg, 1 December 2014.

38. Mandy Visser. Better use your head. How people learn to signal emotions in social contexts. Promotores: M.G.J. Swerts, E.J. Krahmer. Tilburg, 10 June 2015.
39. Sterling Hutchinson. How symbolic and embodied representations work in concert. Promotores: M.M. Louwerse, E.O. Postma. Tilburg, 30 June 2015.
40. Marieke Hoetjes. Talking hands. Reference in speech, gesture and sign. Promotores: E.J. Krahmer, M.G.J. Swerts. Tilburg, 7 October 2015.
41. Elisabeth Lubinga. Stop HIV. Start talking? The effects of rhetorical figures in health messages on conversations among South African adolescents. Promotores: A.A. Maes, C.J.M. Jansen. Tilburg, 16 October 2015.
42. Janet Bagorogoza. Knowledge Management and High Performance. The Uganda Financial Institutions Models for HPO. Promotores: H.J. van den Herik, B. van der Walle, Tilburg, 24 November 2015.
43. Hans Westerbeek. Visual realism: Exploring effects on memory, language production, comprehension, and preference. Promotores: A.A. Maes, M.G.J. Swerts. Co-promotor: M.A.A. van Amelsvoort. Tilburg, 10 Februari 2016.
44. Matje van de Camp. A link to the Past: Constructing Historical Social Networks from Unstructured Data. Promotores: A.P.J. van den Bosch, E.O. Postma. Tilburg, 2 Maart 2016.
45. Annemarie Quispel. Data for all: Data for all: How professionals and non-professionals in design use and evaluate information visualizations. Promotor: A.A. Maes. Co-promotor: J. Schilperoord. Tilburg, 15 Juni 2016.
46. Rick Tillman. Language Matters: The Influence of Language and Language Use on Cognition Promotores: M.M. Louwerse, E.O. Postma. Tilburg, 30 Juni 2016.
47. Ruud Mattheij. The Eyes Have It. Promoteres: E.O. Postma, H. J. Van den Herik, and P.H.M. Spronck. Tilburg, 5 October 2016.
48. Marten Pijl, Tracking of human motion over time. Promotores: E. H. L. Aarts, M. M. Louwerse Co-promotor: J. H. M. Korst. Tilburg, 14 December 2016.
49. Yevgen Matusevych, Learning constructions from bilingual exposure: Computational studies of argument structure acquisition. Promotor: A.M. Backus. Co-promotor: A.Alishahi. Tilburg, 19 December 2016.

50. Karin van Nispen. What can people with aphasia communicate with their hands? A study of representation techniques in pantomime and co-speech gesture. Promotor: E.J. Krahmer. Co-promotor: M. van de Sandt-Koenderman. Tilburg, 19 December 2016.
51. Adriana Baltaretu. Speaking of landmarks. How visual information inuences reference in spatial domains. Promotores: A.A. Maes and E.J. Krahmer. Tilburg, 22 December 2016.
52. Mohamed Abbadi. Casanova 2, a domain specific language for general game development. Promotores: A.A. Maes, P.H.M. Spronck and A. Cortesi. Co-promotor: G. Maggiore. Tilburg, 10 March 2017.
53. Shoshannah Tekofsky. You Are Who You Play You Are. Modelling Player Traits from Video Game Behavior. Promotores: E.O. Postma and P.H.M. Spronck. Tilburg, 19 Juni 2017.
54. Adel Alhuraibi, From IT-Business Strategic Alignment to Performance: A Moderated Mediation Model of Social Innovation, and Enterprise Governance of IT. Promotores: H.J. van den Herik and Prof. dr. B.A. Van de Walle. Co-promotor: Dr. S. Ankolekar. Tilburg, 26 September 2017.
55. Wilma Latuny. The Power of Facial Expressions. Promotores: E.O. Postma and H.J. van den Herik. Tilburg, 29 September 2017.
56. Sylvia Huwaë, Different Cultures, Different Selves? Suppression of Emotions and Reactions to Transgressions across Cultures. Promotores: E.J. Krahmer and J. Schaafsma. Tilburg, 11 October, 2017.
57. Mariana Serras Pereira, A Multimodal Approach to Children's Deceptive Behavior. Promotor: M. Swerts. Co-promotor: S. Shahid Tilburg, 10 January, 2018.
58. Emmelyn Croes, Meeting Face-to-Face Online: The Effects of Video-Mediated Communication on Relationship Formation. Promotores: E.J. Krahmer and M. Antheunis. Co-promotor A.P. Schouten. Tilburg, 28 March 2018.
59. Lieke van Maastricht, Second Language Prosody: Intonation and Rhythm in Production and Perception. Promotores: E.J. Krahmer and M. Swerts. Tilburg, 9 May 2018.
60. Nanne van Noord, Learning visual representations of style. Promotores: E.O. Postma and M. Louwerse. Tilburg, 16 May 2018.

61. Ingrid Masson Carro, Handmade: On the Cognitive Origins of Gestural Representations. Promotores: E.J. Krahmer and M. Goudbeek. Tilburg, 25 June 2018.
62. Bart Joosten, Detecting Social Signals with Spatiotemporal Gabor Filters. Promotores: E.J. Krahmer and E.O. Postma. Tilburg, 29 June 2018.
63. Yan Gu, Chinese hands of time: The effects of language and culture on temporal gestures and spatio-temporal reasoning. Promotor: M. Swerts. Co-promotores: M.W. Hoetjes, R. Cozijn. Tilburg, 5 June 2018.
64. Thiago Castro Ferreira, Advances in Natural Language Generation: Generating Varied Outputs from Semantice Inputs. Promotor: E.J. Krahmer. Co-promotor: S. Wubben. Tilburg, 19 September 2018.