

Survey of the State of the Art in Natural Language Generation: Core tasks, applications and evaluation

Albert Gatt

Institute of Linguistics, University of Malta,

Tal-Qroqq, Msida MSD2080, Malta

albert.gatt@um.edu.mt

Emiel Krahmer

Tilburg center for Cognition and Communication (TiCC),

Tilburg University, P.O.Box 90153, NL-5000 LE,

Tilburg, The Netherlands

e.j.krahmer@tilburguniversity.edu

October 8, 2018

Abstract

This paper surveys the current state of the art in Natural Language Generation (NLG), defined as the task of generating text or speech from non-linguistic input. A survey of NLG is timely in view of the changes that the field has undergone over the past decade or so, especially in relation to new (usually data-driven) methods, as well as new applications of NLG technology. This survey therefore aims to (a) give an up-to-date synthesis of research on the core tasks in NLG and the architectures adopted in which such tasks are organised; (b) highlight a number of relatively recent research topics that have arisen partly as a result of growing synergies between NLG and other areas of artificial intelligence; (c) draw attention to the challenges in NLG evaluation, relating them to similar challenges faced in other areas of NLP, with an emphasis on different evaluation methods and the relationships between them.

Contents

1	Introduction	4
1.1	What is Natural Language Generation?	6
1.2	Why a survey on Natural Language Generation?	7
1.3	Goals of this survey	8
2	NLG Tasks	9
2.1	Content determination	10
2.2	Text structuring	11
2.3	Sentence aggregation	12
2.4	Lexicalisation	14
2.5	Referring expression generation	15
2.6	Linguistic realisation	18
2.6.1	Templates	19
2.6.2	Hand-coded grammar-based systems	19
2.6.3	Statistical approaches	20
2.7	Discussion	21
3	NLG Architectures and Approaches	21
3.1	Rule-based, modular approaches	22
3.2	Planning-based approaches	25
3.2.1	Planning through the grammar	26
3.2.2	Stochastic planning under uncertainty using Reinforcement Learning	28
3.3	Data-driven approaches	29
3.3.1	Acquiring data	30
3.3.2	NLG based on language models	31
3.3.3	NLG as classification and optimisation	33
3.3.4	NLG as ‘parsing’	35
3.3.5	Deep learning methods	37
3.4	Discussion	39
4	The vision-language interface: Image captioning and beyond	40
4.1	Data	41
4.2	The core tasks	42
4.2.1	Image analysis	42
4.2.2	Text generation or retrieval	43
4.3	How is language grounded in visual data?	45
4.4	Vision and language: Current and future directions for NLG	46
5	Variation: Generating text with style, personality and affect	47
5.1	Generating with style: textual variation and personality	48
5.2	Generating with feeling: affect and politeness	50
5.3	Style and affect: concluding remarks	52

6 Generating creative and entertaining text	53
6.1 Generating puns and jokes	54
6.2 Generating metaphors and similes	55
6.3 Generating narratives	57
6.4 Generating creative language: Concluding remarks	61
7 Evaluation	61
7.1 Intrinsic methods	63
7.1.1 Subjective (human) judgements	63
7.1.2 Objective humanlikeness measures using corpora	65
7.1.3 Evaluating genre compatibility	67
7.2 Extrinsic evaluation methods	68
7.3 Black box vs glass box evaluation	69
7.4 On the relationship between evaluation methods	70
7.4.1 Metrics versus human judgements	70
7.4.2 Using controlled experiments	73
7.5 Evaluation: Concluding remarks	73
8 Discussion and future directions	74
8.1 Why (and how) should NLG be used?	75
8.2 NLG isn't about text-to-text...or is it?	75
8.3 Theories and models in search of applications?	76
8.4 Where do we go from here?	77
9 Conclusion	79
References	79

1 Introduction

In his intriguing story The Library of Babel (*La biblioteca de Babel*, 1941), Jorge Luis Borges describes a library in which every conceivable book can be found. It is probably the wrong question to ask, but readers cannot help wondering: who wrote all these books? Surely, this could not be the work of human authors? The emergence of automatic text generation techniques in recent years provides an interesting twist to this question. Consider Philip M. Parker, who offered more than 100.000 books for sale via Amazon.com, including for example his *The 2007-2012 Outlook for Tufted Washable Scatter Rugs, Bathmats, and Sets That Measure 6-Feet by 9-Feet or Smaller in India* Obviously, Parker did not write these 100,000 books by hand. Rather, he used a computer program that collects publicly available information, possibly packaged in human-written texts, and compiles these into a book. Just like the library of Babel contains many books that are unlikely to appeal to a broad audience, Parker's books need not find many readers. In fact, even if only a small percentage of his books get sold a few times, this would still make him a sizeable profit.

Parker's algorithm can be seen to belong to a research tradition of so-called **text-to-text generation** methods, applications that take existing texts as their input, and automatically produce a new, coherent text as output. Other example applications that generate new texts from existing (usually human-written) text include:

- fusion and summarization of related sentences or texts to make them more concise (e.g., Clarke & Lapata, 2010);
- simplification of complex texts, for example to make them more accessible for low-literacy readers (e.g., Siddharthan, 2014) or for children (Macdonald & Siddharthan, 2016);
- automatic spelling, grammar and text correction (e.g., Kukich, 1992; Dale et al., 2012);
- automatic generation of peer reviews for scientific papers (Bartoli et al., 2016);
- generation of paraphrases of input sentences (e.g., Bannard & Callison-Burch, 2005; Kauchak & Barzilay, 2006); and
- automatic generation of questions, for educational and other purposes (e.g., Brown et al., 2005; Rus et al., 2010).

Often, however, it is necessary to generate texts which are not grounded in existing ones. Consider, as a case in point, the minor earthquake that took place close to Beverly Hills, California on March 17, 2014. The Los Angeles Times was the first newspaper to report it, within 3 minutes of the event, providing details about the time, location and strength of the quake. This report was automatically generated by a ‘robo-journalist’, which converted the

incoming automatically registered earthquake data into a text, by filling gaps in a predefined template text.¹

Robo-journalism and associated practices, such as data journalism, are simple examples of what is usually referred to as **data-to-text generation**. They have had a considerable impact in the fields of journalism and media studies (van Dalen, 2012; Clerwall, 2014; Hermida, 2015). The technique used by the Los Angeles Times was not new; many applications have been developed over the years which automatically generate text from non-linguistic data including, but not limited to, systems which produce:

- soccer reports (e.g., Theune et al., 2001; Chen & Mooney, 2008);
- virtual ‘newspapers’ from sensor data (Molina et al., 2011);
- textual descriptions of the day-to-day lives of birds based on satellite data (Siddharthan et al., 2013);
- weather and financial reports (Goldberg et al., 1994; Reiter et al., 2005; Turner et al., 2008; Ramos-Soto et al., 2015; Wanner et al., 2015; Plachouras et al., 2016);
- summaries of patient information in clinical contexts (Hüske-Kraus, 2003; Harris, 2008; Portet et al., 2009; Gatt et al., 2009; Banaee et al., 2013);
- interactive information about cultural artefacts, for example in a museum context (e.g., O’Donnell, 2001; Stock et al., 2007); and
- text intended to persuade (Carenini & Moore, 2006) or motivate behaviour modification (Reiter et al., 2003).

These systems may differ considerably in the quality and variety of the texts they produce, their commercial viability and the sophistication of the underlying methods, but all are examples of **data-to-text generation**. Many of the systems mentioned above focus on imparting information to user. On the other hand, as shown by the examples cited above of systems focussed on persuasion or behaviour change, informing need not be the exclusive goal of NLG. Nor is it a trivial goal in itself, since in order to successfully impart information, a system needs to select what to say, distinguishing it from what can be easily inferred (possibly also depending on the target user), before expressing it coherently.

Generated texts need not have a large audience. There is no need to automatically generate a report of, say, the Champions League European football final, which is covered by many of the best journalists in the field anyway. However, there are many other games, less important to the general public (but presumably very important to the parties involved). Typically, all sports statistics (who played?, who scored? etc.) for these games are stored, but such statistics are not

¹See http://www.slate.com/blogs/future_tense/2014/03/17/quakebot_los_angeles-times_robot_journalist_writes_article_on_la_earthquake.html.

as a rule perused by sport-reporters. Companies like Narrative Science² fill this niche by automatically generating sport reports for these games. Automated Insights³ even generates reports based on user-provided ‘fantasy football’ data. In a similar vein, the automatic generation of weather forecasts for offshore oil platforms (Sripada et al., 2003), or from sensors monitoring the performance of gas turbines (Yu et al., 2006), has proven to be a fruitful application of data-to-text techniques. Such bespoke applications are now the mainstay of companies like Arria-NLG.⁴

Taking this idea one step further, data-to-text generation paves the way for tailoring texts to specific audiences. For example, data from babies in neonatal care can be converted into text differently, with different levels of technical detail and explanatory language, depending on whether the intended reader is a doctor, a nurse or a parent (Mahamood & Reiter, 2011). One could also easily imagine that different sport reports are generated for fans of the respective teams; the winning goal of one team is likely to be considered a lucky one from the perspective of the losing team, irrespective of its ‘objective’ qualities. A human journalist would not dream of writing separate reports about a sports match (if only for lack of time), but for a computer this is not an issue and this is likely to be appreciated by a reader who receives a more personally appropriate report.

1.1 What is Natural Language Generation?

Both text-to-text generation and data-to-text generation are instances of **Natural Language Generation** (NLG). In the most widely-cited survey of NLG methods to date (Reiter & Dale, 1997, 2000), NLG is characterized as ‘the sub-field of artificial intelligence and computational linguistics that is concerned with the construction of computer systems than can produce understandable texts in English or other human languages from some underlying non-linguistic representation of information’ (Reiter & Dale, 1997, p.1). Clearly this definition fits data-to-text generation better than text-to-text generation, and indeed Reiter and Dale (2000) focus exclusively on the former, helpfully and clearly describing the rule-based approaches that dominated the field at the time.

It has been pointed out that precisely defining NLG is rather difficult (e.g., Evans et al., 2002): everybody seems to agree on what the output of an NLG system should be (text), but what the exact input is can vary substantially (McDonald, 1993). A further complication is that the boundaries between different approaches are themselves blurred. For example, text summarisation was characterized above as a text-to-text application; this is clear for so called ‘extractive’ summarizers (which produce summaries using sentences from source documents). However, ‘abstractive’ summarizers (which generate sentences *not* present in any of the source documents) increasingly rely on techniques which are also used in data-to-text, as when opinions are extracted from reviews and

²<https://www.narrativescience.com>

³<https://automatedinsights.com>

⁴<http://www.arria.com>

expressed in completely new sentences (e.g., Labbé & Portet, 2012). Conversely, a data-to-text generation system could conceivably rely on text-to-text generation techniques for learning how to express pieces of data in different or creative ways (McIntyre & Lapata, 2009; Gatt et al., 2009; Kondadadi et al., 2013).

Considering other applications of NLG similarly highlights how blurred boundaries can get. For example, the generation of spoken utterances in dialogue systems (e.g., Walker et al., 2007a; Rieser & Lemon, 2009; Dethlefs, 2014) is another application of NLG, but typically it is closely related to dialogue management, so that management and realisation policies are sometimes learned in tandem (e.g., Rieser & Lemon, 2011b). Even what constitutes ‘a non-linguistic representation of information’ in the context of data-to-text is subject to change: traditionally, this was taken to be database or logically structured information, but in recent times there has been an increased interest in using visual data as input, resulting in so-called **vision-to-text** systems which automatically produce descriptions of static or moving images based on computer vision input (e.g., Mitchell et al., 2012; Kulkarni et al., 2013; Thomason et al., 2014, among many others).

1.2 Why a survey on Natural Language Generation?

Arguably Reiter and Dale (2000) is still the most complete available survey of NLG and the most cited. However, the field of NLG has changed drastically in the last 15 years, with the emergence of successful applications generating tailored reports for specific audiences, and with the emergence of text-to-text as well as vision-to-text generation applications, which also tend to rely more on statistical methods than traditional data-to-text. None of these are covered in Reiter and Dale (2000). Also notably absent are discussions of applications that move beyond standard, ‘factual’ text generation, such as those that account for personality and affect, or creative text such as metaphors and narratives. Finally, a striking omission by Reiter and Dale (2000) is the lack of discussion of evaluation methodology. Indeed, evaluation of NLG output has only recently started to receive systematic attention, in part due to a number of shared tasks that were conducted within the NLG community.

Since Reiter and Dale (2000), various other NLG overview texts have also appeared. Bateman and Zock (2005) covers the cognitive, social and computational dimensions of NLG. McDonald (2010) offers a general characterization of NLG as ‘the process by which thought is rendered into language’ (p. 121). Wanner (2010) zooms in on automatic generation of reports, while Di Eugenio and Green (2010) looks at specific applications, especially in education and health-care. Various specialized collections of articles have also been published, including Krahmer and Theune (2010), which targets data-driven approaches; and Bangalore and Stent (2014) which focusses on interactive systems. The web offers various unpublished technical reports, such as Theune (2003), which surveys dialogue systems, and Piwek (2003) and Belz (2003) on affective NLG. While useful, these resources do not discuss recent developments or offer a comprehensive review. This indicates that a new state-of-the-art survey is highly

timely.

1.3 Goals of this survey

The goal of the current paper is to present a comprehensive overview of NLG developments since 2000, both in order to provide NLG researchers with a synthesis and pointers to relevant research, and to introduce the field to researchers who are less familiar with NLG. Though NLG has been a part of AI and NLP from the early days (see e.g., Winograd, 1972; Appelt, 1985), as a field it has arguably not been fully embraced by these broader communities, and has only recently began to take full advantage of recent advances in data-driven, machine learning and deep learning approaches.

As in Reiter and Dale (2000), our main focus, especially in the first part of the survey, will be on data-to-text generation. In any case, doing full justice to recent developments in the various text-to-text generation applications is beyond the scope of a single survey, and many of these are covered in other individual surveys, including Mani (2001) and Nenkova and McKeown (2011) for summarisation; Androutsopoulos and Malakasiotis (2010) for paraphrasing; and Piwek and Boyer (2012) for automatic question generation. However, we will in various places discuss connections between data-to-text and text-to-text generation, both because – as noted above – the boundaries are blurred, but also, and perhaps more importantly, because text-to-text systems have long been couched in the data-driven frameworks that are becoming increasingly popular in data-to-text generation, also giving rise to some hybrid systems that combine rule-based and statistical techniques (e.g., Kondadadi et al., 2013).

Our review will start with an updated overview of the core NLG tasks that were introduced in Reiter and Dale (2000), followed by a discussion of architectures and approaches, where we pay special attention to those not covered in the Reiter and Dale (2000) survey. These two sections constitute the ‘core’ part of the survey. Beyond these, we highlight several new developments, including approaches where the input data is visual; and research aimed at generating more varied, engaging or creative and entertaining texts, taking NLG beyond the factual, repetitive texts it is sometimes accused of producing. We believe that these applications are not only interesting in themselves, but may also inform more ‘utility’-driven text generation application. For example, by including insights from narrative generation we may be able to generate more engaging reports and by including insights from metaphor generation we may be able to phrase information in these reports in a more original manner. Finally, we will discuss recent developments in evaluation of natural language generation applications.

In short, the goals of this survey are:

- To give an up-to-date synthesis of research on the core tasks in NLG, as well as the architectures adopted in the field, especially in view of recent developments exploiting data-driven techniques (Sections 2 and 3);
- To highlight a number of relatively recent research issues that have arisen partly as a result of growing synergies between NLG and other areas of

artificial intelligence, such as computer vision, stylistics and computational creativity (Sections 4, 5 and 6);

- To draw attention to the challenges in NLG evaluation, relating them to similar challenges faced in other areas of NLP, with an emphasis on different evaluation methods and the relationships between them (Section 7).

2 NLG Tasks

Traditionally, the NLG problem of converting input data into output text was addressed by splitting it up into a number of subproblems. The following six are frequently found in many NLG systems (Reiter & Dale, 1997, 2000); their role is illustrated in Figure 1:

1. *Content determination*: Deciding which information to include in the text under construction,
2. *Text structuring*: Determining in which order information will be presented in the text,
3. *Sentence aggregation*: Deciding which information to present in individual sentences,
4. *Lexicalisation*: Finding the right words and phrases to express information,
5. *Referring expression generation*: Selecting the words and phrases to identify domain objects,
6. *Linguistic realisation*: Combining all words and phrases into well-formed sentences.

These tasks could be thought of in terms of early decision processes (which information to convey to the reader?) to late ones (which words to use in a particular sentence, and how to put them in their correct order?). This characterization reflects a long-running distinction in NLG between *strategy* and *tactics* (a distinction that goes back at least to Thompson, 1977). This distinction also suggests a temporal order in which the tasks are executed, at least in systems with a modular, pipeline architecture (discussed in Section 3.1): for example, the system first needs to decide which input data to express in the text, before it can order information for presentation. However, such ordering of modules is nowadays increasingly put into question in the data-driven architectures discussed below (Section 3). Here, we refer to ‘early’ and ‘late’ tasks by way of distinguishing between choices that are more oriented towards the data (such as what to say) and choices that are of an increasingly linguistic nature (e.g., lexicalisation, or realisation).

In this section, we briefly describe these six tasks, illustrating them with examples, and highlight recent developments in each case. As we shall see,

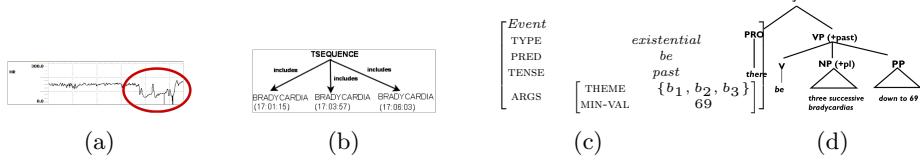


Figure 1: Tasks in NLG, illustrated with a simplified example from the neonatal intensive care domain. First the system has to decide what the important events are in the data (a, content determination), in this case, occurrences of low heart rate (bradycardias). Then it has to decide in which order it wants to present data to the reader (b, text structuring) and how to express these in individual sentence plans (c, aggregation, lexicalisation, reference). Finally, the resulting sentences are generated (d, linguistic realisation).

while the ‘early’ tasks are crucial for the development of NLG systems, they are often intimately connected to the specific application. By contrast, ‘late’ tasks are more often investigated independently of an application, and hence have resulted in approaches that can be shared between applications.

2.1 Content determination

As a first step in the generation process, the NLG system needs to decide which information should be included in the text under construction, and which should not. Typically, more information is contained in data than we want to convey through text, or the data is more detailed than we care to express in text. This is clear in Figure 1a, where the input signal – a patient’s heart rate – only contains a few patterns of interest. Selection may also depend on the target audience (e.g. does it consist of *experts* or *novices*, for example) and on the overall communicative intention (e.g. should the text *inform* the reader or *convince* him to do something).

Content determination involves choice. In a soccer report, we may not want to verbalise each pass and foul committed, even though the data may contain this information. In the case of neonatal care, data might be collected continuously from sensors measuring heart rate, blood pressure and other physiological parameters. Data thus needs to be filtered and abstracted into a set of *preverbal messages*, semantic representations of information which are often expressed in a formal representation language, such as logical or database languages, attribute-value matrices or graph structures. They can express, among other things, which relations hold between which domain entities, for example, expressing that player X scored the first goal for team Y at time T.

Though content determination is present in most NLG systems (cf. Mellish et al., 2006), approaches are typically closely related to the domain of applica-

tion. A notable exception is Guhe (2007), an incremental account of content determination based on studies of speakers' descriptions of dynamic events as they unfold. This work belongs to a strand of research which considers NLG first and foremost as a methodology eminently suitable for understanding *human* language production.

In recent years, researchers have started exploring data-driven techniques for content determination ((see e.g., Barzilay & Lee, 2004; Bouayad-Agha et al., 2013; Kutlak et al., 2013; Venigalla & Di Eugenio, 2013). Barzilay and Lee (2004), for example, used hidden markov models to model topic shifts in a particular domain of discourse (say, earthquake reports), where the hidden states represented 'topics', modelled as sentences clustered together by similarity. A clustering approach was also used by Duboue and McKeown (2003) in the biography domain, using texts paired with a knowledge base, from which semantic data was clustered and scored according to its occurrence in text. In a similar vein Barzilay and Lapata (2005) use a database of American football records and corresponding text. Their aim was not only to identify bits of information that should be mentioned, but also dependencies between them, since mentioning a certain event (say, a score by a quarterback) may warrant the mention of another (say, another scoring event by a second quarterback). The solution proposed by Barzilay and Lapata was to compute both individual preference scores for events, and a link preference score.

More recently, various researchers have addressed the question of how to automatically learn alignments between data and text, also in the broader context of grounded language acquisition, i.e., modelling how we learn language by looking at correspondences between objects and events in the world and the way we refer to them in language (Roy, 2002; Yu & Ballard, 2004; Yu & Siskind, 2013). For example, Liang et al. (2009) extended the work by Barzilay and Lapata (2005) to multiple domains (soccer and weather), relying on weakly supervised techniques; in a similar vein, Koncel-Kedziorski et al. (2014) presented a weakly supervised multilevel approach, to deal with the fact that there is no one-to-one correspondence between, for example, soccer events in data and sentences in associated soccer reports. We shall return to these methods as part of a broader discussion of data-driven approaches below (Section 3.3).

2.2 Text structuring

Having determined what messages to convey, the NLG system needs to decide on their order of presentation to the reader. For example, Figure 1b shows three events of the same type (all bradycardia events, that is, brief drops in heart rate), selected (after abstraction) from the input signal and ordered as a temporal sequence.

This stage is often referred to as text (or discourse or document) structuring. In the case of the soccer domain, for example, it seems reasonable to start with general information (where and when the game was played, how many people attended, etc.), before the goals are described, typically in temporal order. In the neonatal care domain, a temporal order can be imposed among specific

events, as in Figure 1b, but larger spans of text may reflect ordering based on importance, and grouping of information based on relatedness (e.g. all events related to a patient’s respiration) (Portet et al., 2009). Naturally, alternative discourse relations may exist between separate messages, such as *contrasts* or *elaborations*. The result of this stage is a discourse, text or document plan, which is a structured and ordered representation of messages.

These examples again imply that the application domain imposes constraints on ordering preferences. Early approaches, such as McKeown (1985), often relied on hand-crafted, domain-dependent structuring rules (which McKeown called *schemata*). To account for discourse relations between messages, researchers have alternatively relied on Rhetorical Structure Theory (RST; e.g., Mann & Thompson, 1988; Scott & Sieckenius de Souza, 1990; Hovy, 1993), which also typically involved domain-specific rules. For example, Williams and Reiter (2008) used RST relations to identify ordering among messages that would maximise clarity to low-skilled readers.

Various researchers have explored the possibilities of using machine learning techniques for document structuring (e.g., Dimitromanolaki & Androutsopoulos, 2003), sometimes doing this in tandem with content selection (Duboue & McKeown, 2003). General approaches for information ordering (Barzilay & Lee, 2004; Lapata, 2006) have been proposed, which automatically try to find an optimal ordering of ‘information-bearing items’. These approaches can be applied to text structuring, where the items to be ordered are typically preverbal messages; however, they can also be applied in (multidocument) summarisation, where the items to be ordered are sentences from the input documents which are judged to be summary-worthy enough to include (e.g., Barzilay et al., 2002; Bollegala et al., 2010).

2.3 Sentence aggregation

Not every message in the text plan needs to be expressed in a separate sentence; by combining multiple messages into a single sentence, the generated text becomes potentially more fluid and readable (e.g., Dalianis, 1999; Cheng & Mellish, 2000), although there are also situations where it has been argued that aggregation should be avoided (discussed in Section 5.2). For instance, the three events selected in Figure 1b are shown as ‘merged’ into a single pre-linguistic representation, which will be mapped to a single sentence. The process by which related messages are grouped together in sentences is known as sentence aggregation.

To take another example, from the soccer domain, one (unaggregated) way to describe the fastest hat-trick in the English Premier League would be:

- (1) Sadio Mane scored for Southampton after 12 minutes and 22 seconds.
- (2) Sadio Mane scored for Southampton after 13 minutes and 46 seconds.
- (3) Sadio Mane scored for Southampton after 15 minutes and 18 seconds.

Clearly, this is rather redundant, not very concise or coherent, and generally unpleasant to read. An aggregated alternative, such as the following, would therefore be preferred:

- (4) Sadio Mane scored three times for Southampton in less than three minutes.

In general, aggregation is difficult to define, and has been interpreted in various ways, ranging from redundancy elimination to linguistic structure combination. Reape and Mellish (1999) offer an early survey, distinguishing between aggregation at the semantic level (as illustrated in Figure 1c) and at the level of syntax, illustrated in the transition from (2.3) to (4) above.

It is probably fair to say that much early work on aggregation was strongly domain-dependent. This work focussed on domain- and application-specific rules (e.g. ‘if a player scores two consecutive goals, describe these in the same sentence’), that were typically hand-crafted (e.g., Hovy, 1988; Dalianis, 1999; Shaw, 1998). Once again, more recent work has evinced a turn towards data-driven approaches, where aggregation rules are acquired from corpus data (e.g., Walker et al., 2001; Cheng & Mellish, 2000). Barzilay and Lapata (2006) present a system that learns how to aggregate on the basis of a parallel corpus of sentences and corresponding database entries, by looking for similarities between entries. As was the case with the content selection method of Barzilay and Lapata (2005), Barzilay and Lapata (2006) view the problem in terms of global optimisation: an initial classification is done over pairs of database entries which determines whether they should be aggregated or not on the basis of their pairwise similarity. Subsequently, a globally optimal set of linked entries is selected based on transitivity constraints (if $\langle e_i, e_j \rangle$ and $\langle e_j, e_k \rangle$ are linked, then so should $\langle e_i, e_k \rangle$) and global constraints, such as how many sentences should be aggregated in a document. Global optimisation is cast in terms of Integer Linear Programming, a well-known mathematical optimization technique (e.g., Nemhauser & Wolsey, 1988).

With syntactic aggregation, it is arguably more feasible to define domain-independent rules to eliminate redundancy (Harbusch & Kempen, 2009; Kempen, 2009). For example, converting the first example into the second below

- (5) Sadio Mane scored in the 12th minute and he scored again in the 13th minute.
- (6) Sadio Mane scored in the 12th minute and again in the 13th.

could be achieved by identifying the parallel verb phrases in the two conjoined sentences and eliding the subject and verb in the second. Recent work has explored the possibility of acquiring such rules from corpora automatically. For example, Stent and Molina (2009) describe an approach to the acquisition of sentence-combining rules from a discourse treebank, which are then incorporated into the SPAKY sentence planner described by Walker et al. (2007b). A more general approach to the same problem is discussed by White and Howcroft (2015).

Arguably, aggregation on the syntactic level can only account for relatively small reductions, compared to aggregation at the level of messages. Furthermore, syntactic aggregation assumes that the sentence planning process (which includes lexicalisation) is complete. Hence, while traditional approaches to NLG view aggregation as part of sentence planning, which occurs prior to syntactic realisation, the validity of this claim depends on the type of aggregation being performed (see also Theune et al., 2006).

2.4 Lexicalisation

Once the content of the sentence has been finalised, possibly also as a result of aggregation at the message level, the system can start converting it into natural language. In our example (Figure 1c), the outcome of aggregation and lexicalisation are shown together: here, the three events have been grouped, and mapped to a representation that includes a verb (*be*) and its arguments, though the arguments themselves still have to be rendered in a referring expression (see below). This reflects an important decision, namely, which words or phrases to use to express the messages' building blocks. A complication is that often a single event can be expressed in natural language in many different ways. A scoring event in a soccer match, for example, can be expressed as 'to score a goal', 'to have a goal noted', 'to put the ball in the net', among many others.

The complexity of this lexicalisation process critically depends on the number of alternatives that the NLG system can entertain. Often, contextual constraints play an important role here as well: if the aim is to generate texts with a certain amount of variation (e.g., Theune et al., 2001), the system can decide to randomly select a lexicalisation option from a set of alternatives (perhaps even from a set of alternatives not used earlier in the text). However, stylistic constraints come into play: 'to score a goal' is an unfortunate way of expressing an own goal, for example. In other applications, lexical choice may even be informed by other considerations, such as the attitude or affective stance towards the event in question (e.g., Fleischman & Hovy, 2002, and the discussion in Section 5). Whether or not NLG systems aim for variation in their output or not depends on the domain. For example, variation in soccer reports is presumably more appreciated by readers than variation in weather reports (on which see Reiter et al., 2005); it may also depend on where in a text the variation occurs (e.g., variation in expressing timestamps may be less appreciated than variation in referential forms, see e.g., Ferreira et al. (2016)).

One straightforward model for lexicalisation – the one assumed in Figure 1 – is to operate on preverbal messages, converting domain concepts directly into lexical items. This might be feasible in well-defined domains. More often, lexicalisation is harder, for at least two reasons (cf. Bangalore & Rambow, 2000): First, it can involve selection between semantically similar, near-synonymous or taxonomically related words (e.g. *animal* vs *dog*; Stede, 2000; Edmonds & Hirst, 2002). Second, it is not always straightforward to model lexicalisation in terms of a crisp concept-to-word mapping. One source of difficulty is vagueness, which arises, for example, with terms denoting properties that are gradable.

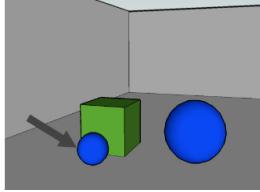
For example, selecting the adjectives ‘wide’ or ‘tall’ based on the dimensions of an entity requires the system to reason about the width or height of similar objects, perhaps using some standard of comparison (since a ‘tall glass’ is shorter than a ‘short man’; cf. Kennedy & McNally, 2005; van Deemter, 2012). A similar issue has been noted in the context of presenting numerical information, such as timestamps and quantities (Reiter et al., 2005; Power & Williams, 2012). For example, Reiter et al. (2005) discussed time expressions in the context of weather-forecast generation, pointing out that a timestamp *00:00* could be expressed as *late evening*, *midnight*, or simply *evening* (Reiter et al., 2005, p. 143). Not surprisingly, humans (including the professional forecasters that contributed to Reiter et al.’s evaluation), show considerable variation in their lexical choices.

It is interesting to note that many issues related to lexicalisation have also been discussed in the psycholinguistic literature on lexical access (Levelt, 1989; Levelt et al., 1999). Among these is the question of how speakers home in on the right word and under what conditions they are liable to make errors, given that the mental lexicon is a densely connected network in which lexical items are connected at multiple levels (semantic, phonological, etc). This has also been a fruitful topic for computational modelling (e.g., Levelt et al., 1999). In contrast to cognitive modelling approaches, however, research in NLG increasingly views lexicalisation as part of surface realisation (discussed below) (a similar observation is made by Mellish & Dale, 1998, p.351). A fundamental contribution in this context is by Elhadad et al. (1997), who describe a unification-based approach, unifying conceptual representations (i.e., preverbal messages) with grammar rules encoding lexical as well as syntactic choices.

2.5 Referring expression generation

Referring Expression Generation (REG) is characterised by Reiter and Dale (1997, p.11) as ‘the task of selecting words or phrases to identify domain entities’. This characterisation suggests a close similarity to lexicalisation, but Reiter and Dale (2000) point out that the essential difference is that referring expression generation is a ‘discrimination task, where the system needs to communicate sufficient information to distinguish one domain entity from other domain entities’. REG is among the tasks within the field of automated text generation that has received most attention in recent years (Mellish et al., 2006; Siddharthan et al., 2011). Since it can be separated relatively easily from a specific application domain and studied in its own right, various ‘standalone’ solutions for the REG problem exist.

In our running example, the three bradycardia events shown in Figure 1b are later represented as a set of three entities under the THEME argument of *be*, following lexicalisation (Figure 1c). How the system refers to them will depend, among other things, on whether they’ve already been mentioned (in which case, a pronoun or definite description might work) and if so, whether they need to be distinguished from any other similar entities (in which case, they might need to be distinguished by some properties, such as the time when they occurred).



(a) Visual domain from the GRE3D corpus (Viethen & Dale, 2008)

Attributes	Domain objects		
	d_1	d_2	d_3
Color	blue	green	blue
Shape	ball	cube	ball
Size	small	large	large
Relation	before(d_2)	behind(d_1)	next_to(d_2)

(b) Tabular representation of the visual domain

Figure 2: Visual domain and tabular representation

The first choice is therefore related to *referential form*: whether entities are referred to using a pronoun, a proper name or an (in)definite description, for example. This depends partly on the extent to which the entity is ‘in focus’ or ‘salient’ (see e.g., Poesio et al., 2004) and indeed such notions underlie many computational accounts of pronoun generation (e.g., McCoy & Strube, 1999; Callaway & Lester, 2002; Kibble & Power, 2004). Choosing referential forms has recently been the topic of a series of shared tasks on the Generation of Referring Expressions in Context (GREC; Belz et al., 2010), using data from Wikipedia articles, which included choices such as reflexive pronouns and proper names. Many systems participating in this challenge framed the problem in terms of classification among these many options. Still, it is probably fair to say that much work on referential form has focussed on when to use pronouns. Forms such as proper names remain understudied, although recently various researchers have highlighted the problems of proper name generation (Siddharthan et al., 2011; van Deemter, 2016; Ferreira et al., 2017).

Determining the *referential content* usually comes into play when the chosen form is a description. Typically, there are multiple entities which have the same referential category or type in a domain (more than one player, for example, or several bradycardias). As a result, other properties of the entity will need to be mentioned if it is to be identified by the reader or hearer. Earlier REG research often worked with simple visual domains, such as Figure 2a or its corresponding tabular representation, taken from the GRE3D corpus (Viethen & Dale, 2008). In this example, the REG content selection problem is to find a set of properties for a target (say d_1) that singles it out from its two distractors (d_2 and d_3).

REG content determination algorithms can be thought of as performing a search through the known properties of the referent for the ‘right’ combination that will distinguish it in context. What constitutes the ‘right’ combination depends on the underlying theory. Too much information in the description (as in *the small blue ball before the large green cup*) might be misleading or even boring; too little (*the ball*) might hinder identification. Much work on REG has

appealed to the Gricean maxim stating that speakers should make sure that their contributions are sufficiently informative for the purposes of the exchange, but not more so (Grice, 1975). How this is interpreted has been the subject of a number of algorithmic interpretations, including:

- Conducting an exhaustive search through the space of possible descriptions and choosing the smallest set of properties that will identify the target referent, the strategy incorporated by the Full Brevity procedure (Dale, 1989). In our example domain, this would select size.
- Selecting properties incrementally, but choosing the one which rules out most distractors at each step, thereby minimising the possibility of including information that isn't directly relevant to the identification task. This is the underlying idea of the Greedy Heuristic algorithm (Dale, 1989, 1992), and it has more recently been revived in stochastic utility-based models such as Frank et al. (2009). In our example scene, such an algorithm would once again consider size first.
- Selecting properties incrementally, but based on domain-specific preference or cognitive salience. This is the strategy incorporated in the Incremental Algorithm (Dale & Reiter, 1995), which would predict that color should be preferred over size in our example.

While these heuristics focus exclusively on the requirement that a referent be unambiguously identified, research on reference in dialogue (e.g., Jordan & Walker, 2005) has shown that under certain conditions, referring expressions may also include 'redundant' properties in order to achieve other communicative goals, such as confirmation of a previous utterance by an interlocutor. Similarly, White et al. (2010) present a system which generates user-tailored descriptions in spoken dialogue, arguing that, for example, a frequent flyer would prefer different descriptions of flights than a student who only flies occasionally.

These various algorithms compute (possibly different) distinguishing descriptions for target referents (more precisely: they select sets of properties that distinguish the target, but that still need to be expressed in words; see Section 2.6 below). Various strands of more recent work can be distinguished (surveyed in Krahmer & van Deemter, 2012). Some researchers have focussed on extending the expressivity of the 'classical' algorithms, to include plurals (*the two balls*) and relations (*the ball in front of a cube*) (e.g., Horacek, 1997; Stone, 2000; Gardent, 2002; Kelleher & Kruijff, 2006; Viethen & Dale, 2008, among many others). Other work has cast the problem in probabilistic terms; for example, FitzGerald et al. (2013) frame REG as a problem of estimating a log-linear distribution over a space of logical forms representing expressions for sets of objects. Other work has concentrated on evaluating the performance of different REG algorithms, by collecting controlled human references and comparing these with the references predicted by various algorithms (e.g., Belz, 2008; Gatt & Belz, 2010; Jordan & Walker, 2005, again among many others). In a similar vein,

researchers have also started exploring the relevance of REG algorithms as psycholinguistic models of human language production (e.g., van Deemter et al., 2012b).

A different line of work has moved away from the separation between content selection and form, performing these tasks jointly. For example, Engonopoulos and Koller (2014) use a synchronous grammar that directly relates surface strings to target referents, using a chart to compute the possible expressions for a given target. This work bears some relationship to planning-based approaches we discuss in Section 3.2 below, which exploit grammatical formalisms as planning operators (e.g. Stone & Webber, 1998; Koller & Stone, 2007), solving realisation and content determination problems in tandem (including REG as a special case).

Finally, in earlier work visual information was typically ‘simplified’ into a table (as we did above), but there has been substantial progress on REG in more complex scenarios. For example, the GIVE challenge (Koller et al., 2010), provided impetus for the exploration of situated reference to objects in a virtual environment (see also Stoia & Shockley, 2006; Garoufi & Koller, 2013). More recent work has started exploring the interface between computer vision and REG to produce descriptions of objects in complex, realistic visual scenes, including photographs (e.g., Mitchell et al., 2013; Kazemzadeh et al., 2014; Mao et al., 2016). This forms part of a broader set of developments focussing on the relationship between vision and language, which we turn to in Section 4.

2.6 Linguistic realisation

Finally, when all the relevant words and phrases have been decided upon, these need to be combined to form a well-formed sentence. The simple example in Figure 1d shows the structure underlying the sentence *there were three successive bradycardias down to 69*, the linguistic message corresponding to the portion selected from the original signal in Figure 1a.

Usually referred to as linguistic realisation, this task involves ordering constituents of a sentence, as well as generating the right morphological forms (including verb conjugations and agreement, in those languages where this is relevant). Often, realisers also need to insert function words (such as auxiliary verbs and prepositions) and punctuation marks. An important complication at this stage is that the output needs to include various linguistic components that may not be present in the input (an instance of the ‘generation gap’ discussed in Section 3.1 below); thus, this generation task can be thought of in terms of projection between non-isomorphic structures (cf. Ballesteros et al., 2015). Many different approaches have been proposed, of which we will discuss

1. human-crafted templates;
2. human-crafted grammar-based systems;
3. statistical approaches.

2.6.1 Templates

When application domains are small and variation is expected to be minimal, realisation is a relatively easy task, and outputs can be specified using templates (e.g., Reiter et al., 1995; McRoy et al., 2003), such as the following.

- (7) \$player scored for \$team in the \$minute minute.

This template has three variables, which can be filled with the names of a player, a team, and the minute in which this player scored a goal. It can thus serve to generate sentences like:

- (8) Ivan Rakitic scored for Barcelona in the 4th minute.

An advantage of templates is that they allow for full control over the quality of the output and avoid the generation of ungrammatical structures. Modern variants of the template-based method include syntactic information in the templates, as well as sophisticated rules for filling the gaps (Theune et al., 2001), making it difficult to distinguish templates from more sophisticated methods (van Deemter et al., 2005). The disadvantage of templates is that they are labour-intensive if constructed by hand (though templates have recently been learned automatically from corpus data, see e.g., Angeli et al., 2012; Kondadadi et al., 2013, and the discussion in Section 3.3 below). They also do not scale well to applications which require considerable linguistic variation.

2.6.2 Hand-coded grammar-based systems

An alternative to templates is provided by general-purpose, domain-independent realisation systems. Most of these systems are *grammar-based*, that is, they make some or all of their choices on the basis of a grammar of the language under consideration. This grammar can be manually written, as in many classic off-the-shelf realisers such as FUF/SURGE (Elhadad & Robin, 1996), MUMBLE (Meteer et al., 1987), KPML (Bateman, 1997), NIGEL (Mann & Matthiessen, 1983), and RealPro (Lavoie & Rambow, 1997). Hand-coded grammar-based realisers tend to require very detailed input. For example, KPML (Bateman, 1997) is based on Systemic-Functional Grammar (SFG; Halliday & Matthiessen, 2004), and realisation is modelled as a traversal of a network in which choices depend on both grammatical and semantico-pragmatic information. This level of detail makes these systems difficult to use as simple ‘plug-and-play’ or ‘off the shelf’ modules (e.g., Kasper, 1989), something which has motivated the development of simple realisation engines which provide syntax and morphology APIs, but leave choice-making up to the developer (Gatt et al., 2009; Vaudry & Lapalme, 2013; Bollmann, 2011; de Oliveira & Sripada, 2014).

One difficulty for grammar-based systems is how to make choices among related options, such as the following, where hand-crafted rules with the right sensitivity to context and input are difficult to design:

- (9) Ivan Rakitic scored for Barcelana in the 4th minute.

- (10) For Barcelona, Ivan Rakitic scored in minute four.
- (11) Barcelona player Ivan Rakitic scored after four minutes.

2.6.3 Statistical approaches

Recent approaches have sought to acquire probabilistic grammars from large corpora, cutting down on the amount of manual labour required, while increasing coverage. Essentially, two approaches have been taken to include statistical information in the realisation process. One approach, introduced by the seminal work of Langkilde and Knight (Langkilde-Geary, 2000; Langkilde-Geary & Knight, 2002) on the HALOGEN/NITROGEN systems, relies on a two-level approach, in which a small, hand-crafted grammar is used to generate alternative realisations represented as a forest, from which a stochastic re-ranker selects the optimal candidate. Langkilde and Knight rely on corpus-based statistical knowledge in the form of n-grams, whereas others have experimented with more sophisticated statistical models to perform reranking (e.g., Bangalore & Rambow, 2000; Ratnaparkhi, 2000; Cahill et al., 2007). The second approach does not rely on a computationally expensive generate-and-filter approach, but uses statistical information directly at the level of generation decisions. An example of this approach is the pCRU system developed by Belz (2008), which generates the most likely derivation of a sentence, given a corpus, using a context-free grammar. In this case, the statistics are exploited to control the generator's choice-making behaviour as it searches for the optimal solution.

In both approaches, the base generator is hand-crafted, while statistical information is used to filter outputs. An obvious alternative would be to *also* rely on statistical information for the base-generation system. Fully data-driven grammar-based approaches have been developed by acquiring grammatical rules from treebanks. For example, the OpenCCG framework (Espinosa et al., 2008; White & Rajkumar, 2009, 2012) presents a broad coverage English surface realizer, based on Combinatory Categorial Grammar (CCG; Steedman, 2000), relying on a corpus of CCG representations derived from the Penn Treebank (Hockenmaier & Steedman, 2007) and using statistical language models for reranking. There are several other approaches to realisation that adopt a similar rationale, based on a variety of grammatical formalisms, including Head-Driven Phrase Structure Grammar (HPSG; Nakanishi et al., 2005; Carroll & Oepen, 2005), Lexical-Functional Grammar (LFG; Cahill & Josef, 2006) and Tree Adjoining Grammar (TAG; Gardent & Narayan, 2015). In the many of these systems, the base generator uses some variant of the chart generation algorithm (Kay, 1996) to iteratively realise parts of an input specification and merge them into one or more final structures, which can then be ranked (see Rajkumar & White, 2014, for further discussion). The existence of stochastic realisers with wide-coverage grammars has motivated a greater focus on subtle choices, such as how to avoid structural ambiguity, or how to handle choices such as explicit complementiser insertion in English (see e.g., Rajkumar & White, 2011).

Other approaches to realisation also rely on one or more classifiers to im-

prove outputs. For example, Filippova and Strube describe an approach to linearisation of constituents using a two-step approach with Maximum Entropy classifiers, first determining which constituent should occupy sentence-initial position, then ordering the constituents in the remainder of the sentence (Filippova & Strube, 2007, 2009). Bohnet et al. (2010) describe a realiser using underspecified dependency structures as input, in a framework based on Support Vector Machines, where classifiers are organised in a cascade. An initial classifier decodes semantic input into the corresponding syntactic features, while two subsequent classifiers first linearise the syntax and then render the correct morphological realisation for the component lexemes. This ‘deep generation’ approach was applied to four languages – Chinese, English, German and Spanish – and found to outperform the approach of Filippova and Strube (2009) on English when compared to a corpus using the BLEU metric (Papineni et al., 2002), though it falls somewhat short of the German realiser of Filippova and Strube (2007), where the two-step classification approach does better.

Modelling choices using classifier cascades is not restricted to realisation; indeed, in some cases, it has been adopted as a model for the NLG process as a whole, a topic we will return to in Section 3.3.3. One outcome of this view of NLG is that the nature of the input representation also changes: the more decisions that are made within the statistical generation system, the less linguistic and more abstract the input representation becomes, paving the way for integrated, end-to-end stochastic generation systems, such as Konstas and Lapata (2013), which we also discuss in the next section.

2.7 Discussion

This section has given an overview of some classic tasks that are found in most NLG systems. One of the common trends that can be identified in each case is the steady move from early, hand-crafted approaches based on rules, to the more recent stochastic approaches that rely on corpus data, with a concomitant move towards more domain-independent approaches. Historically, this was the case already for tasks, such as referring expression generation or realisation, which became topics of intensive research in their own right. However, as more and more approaches to all NLG tasks begin to take a statistical turn, there is increasing emphasis on learning techniques; the domain-specific aspect is, as it were, incidental, a property of the training data itself. As we shall see in the next section, this trend has also influenced the way different NLG tasks are organised, that is, the architecture of systems for text generation from data.

3 NLG Architectures and Approaches

Having given an overview of the most common sub-tasks that NLG systems incorporate, we now turn to the way such tasks can be organised. Broadly speaking, we can distinguish between three dominant approaches to NLG architectures:



Figure 3: Classical three-stage NLG architecture, after Reiter and Dale (2000). Darker segments illustrate the three main modules; lighter segments show the outputs.

1. *Modular* architectures: These are often typical of systems with roots in the classical, symbol-processing paradigm that dominated early AI research. By design, such architectures involve fairly crisp divisions among sub-tasks, though with significant variations among them;
2. *Planning* perspectives: Again with deep roots in the AI tradition, viewing text generation as planning affords a more integrated, less modular design;
3. *Data-driven*, integrated approaches: Now the dominant trend in NLG (as it is in NLP more generally), such approaches place a heavy reliance on statistical learning of correspondences between (non-linguistic) inputs and outputs. Such correspondences often cut across task divisions, resulting once again in more integrated approaches to the NLG problem.

Of these three, the first, modular design is the oldest and for a long time, following Reiter (1994), was referred to as the ‘consensus’. While we review it in some depth below, we emphasise that its consensual status has been repeatedly put into question. Indeed, more recent planning-based and/or data-driven research has strongly challenged the modular view. For this reason, in what follows, we will often explicitly contrast the encapsulated design of the older model with these more ‘global’ approaches, with a view to highlighting computational solutions aimed to address NLG sub-tasks jointly.

3.1 Rule-based, modular approaches

Existing surveys of NLG, including Reiter and Dale (Reiter & Dale, 1997, 2000) and Reiter (2010) typically refer to some version of the pipeline architecture displayed in Figure 3 as the ‘consensus’ architecture in the field. Originally introduced by (Reiter, 1994), the pipeline was a generalisation based on actual practice and achieved the status of a ‘de facto standard’.

Different modules in the pipeline incorporate different subsets of the tasks described in Section 2. The first module, the *Text Planner* (or Document Planner, or Macroplanner), combines content selection and text structuring (or document planning). Thus, it is concerned mainly with strategic generation (McDonald,

1993), the choice of ‘what to say’. The resulting text plan, a structured representation of messages, is the input to the *Sentence Planner* (or microplanner), which typically combines sentence aggregation, lexicalisation and referring expression generation (Reiter & Dale, 2000). If text planning amounts to *deciding what to say*, sentence planning can be understood as deciding *how to say it*. All that remains then is to actually say it, i.e., generate the final sentences in a grammatically correct way, by applying syntactic and morphological rules. This task is performed by the *Linguistic Realiser*. Together, sentence planning and realisation encompass the set of tasks traditionally referred to as *tactical generation*.

Interestingly, when Reiter (1994) proposed this three-stage architecture as the emerging consensus architecture in NLG, he drew a parallel with human speech production, where the most influential psycholinguistic model of language production, proposed by Levelt (1989, 1999), makes a similar distinction between deciding what to say and determining how to say it. Levelt’s model allows for a limited degree of self-monitoring through feedback loops, a feature that is absent in Reiter’s NLG pipeline, but continues to play an important role in psycholinguistics (cf. Pickering & Garrod, 2013), though here too there has been increasing emphasis on more integrated models.

The consensus pipeline also shares a number of characteristics with a widely-used architecture in text summarisation (Mani, 2001; Nenkova & McKeown, 2011). Rather like the program used by Parker to generate books (see Section 1), summarisation systems take as input one or more texts, seeking to produce a summary for the reader. Traditionally (as discussed by Mani, 2001, among others), summarisation can be broken down into the following steps:

1. *Analysis* of the source text(s), whereby information – in the form of phrases or sentences – is selected for inclusion in the eventual summary. Since this stage involves selection, it shares some features with the text planning stage of a data-to-text system, where content determination is one of the tasks;
2. *Transformation* of the selected input, where selected phrases or sentences can undergo processes such as aggregation, fusion or paraphrasing to reduce redundancy and make the text fluent. This stage, which is especially important in abstractive summarisation, shares some features with the sentence planning stage in Figure 3;
3. *Synthesis*, that is, the process of generating the summary, based on the selected information. In this case, systems are typically dealing with textual input, but the higher the level of abstraction in the summary, the more this stage will play a role in re-generating text that might look quite different in its essentials from the original input text(s). Hence, this task may share some features with the realisation stage in Figure 3.

A hallmark of the architecture in Figure 3 is that it represents clear-cut divisions among tasks that are traditionally considered to belong to the ‘what’

(strategic) and the ‘how’ (tactical). However, this does not imply that this division is universally accepted in practice. In a survey conducted approximately a decade ago, Mellish et al. (2006) already concluded that while several NLG systems incorporate many of the core tasks outlined in Section 2, their organisation varies considerably from system to system. Indeed, some tasks may be split up across modules. For example, the content determination part of referring expression generation might be placed in the sentence planner, but decisions about form (such as whether to use an anaphoric NP, and if so, what kind of NP to produce) may have to wait until at least some realisation-related decisions have been taken. Based on these observations, Mellish et al. proposed an alternative formalism, the ‘objects-and-arrows’ framework, within which different types of information flow between NLG sub-tasks can be accommodated. Rather than offering a specific architecture, this framework was intended as a formalism within which high-level descriptions of different architectures can be specified. However, it retains the principle that the tasks, irrespective of their organisation, are well-defined and distinguished.

A more recent development in relation to the pipeline architecture in Figure 3 is a proposal by Reiter (2007) to accommodate systems in which input consists of raw (often numeric) data that requires some preprocessing before it can undergo the kind of selection and planning that the Text Planner is designed to execute. The main characteristic of these systems is that input is unstructured, in contrast to systems which operate over logical forms, or database entries. Examples of application domains where this is the case include weather reporting (e.g., Goldberg et al., 1994; Busemann & Horacek, 1997; Coch, 1998; Turner et al., 2008; Sripada et al., 2003; Ramos-Soto et al., 2015), where the input often takes the form of numerical weather predictions; and generation of summaries from patient data (e.g., Hueske-Kraus, 2003; Harris, 2008; Gatt et al., 2009; Banaee et al., 2013). In such cases, NLG systems often need to perform some form of *data abstraction* (for example, identifying broad trends in the data), followed by *data interpretation*. The techniques used to perform these tasks range from extensions of signal processing techniques (e.g., Portet et al., 2009) to the application of reasoning formalisms based on fuzzy set theory (e.g., Ramos-Soto et al., 2015). Reiter (2007)’s proposal accommodates these steps by extending the pipeline ‘backwards’, incorporating stages prior to Text Planning.

Notwithstanding its elegance and simplicity, there are challenges associated with a pipeline NLG architecture, of which two are particularly worth highlighting:

- The *generation gap* (Meteer, 1991) refers to mismatches between strategic and tactical components, so that early decisions in the pipeline have unforeseen consequences further downstream. To take an example from Inui et al. (1992), a generation system might determine a particular sentence ordering during the sentence planning stage, but this might turn out to be ambiguous once sentences have actually been realised and orthography has been inserted;
- *Generating under constraints:* Itself perhaps an instance of the generation

gap, this problem can occur when the output of a system has to match certain requirements, for example, it cannot exceed a certain length (see Reiter, 2000, for discussion). Formalising this constraint might appear possible at the realisation stage – by stipulating the length constraint in terms of number of words or characters, for instance – but it is much harder at the earlier stages, where the representations are pre-linguistic and their mapping to the final text are potentially unpredictable.

These, and related problems, motivated the development of alternative architectures. For instance, some early NLG systems were based on an interactive design, in which a module’s initially incomplete output could be fleshed out based on feedback from a later module (the PAULINE system is an example of this; Hovy, 1988). An even more flexible stance is taken in blackboard architectures, in which task-specific procedures are not rigidly pre-organised, but perform their tasks reactively as the output, represented in a data structure shared between tasks, evolves (e.g., Nirenburg et al., 1989). Finally, revision-based architectures allow a limited form of feedback between modules under monitoring, with the possibility of altering choices which prove to be unsatisfactory (e.g., Mann & Moore, 1981; Inui et al., 1992). This has the advantage of not requiring ‘early’ modules to be aware of the consequences of their choices for subsequent modules, since something that goes wrong can always be revised (Inui et al., 1992). Revision need not be carried out exclusively to rectify shortcomings. For instance, Robin (1993) used revision in the context of sports summaries; an initial draft was revised to add historical background information that was made relevant by the events reported in the draft, also taking decisions as to where to place them in relation to the main text. The price that all of these alternatives potentially incur is, of course, a reduction in efficiency, as noted by De Smedt et al. (1996).

Alternatives to pipelines often end up blurring the boundaries between modules in the NLG system. This is a feature that is even more evident in some planning-based and data-driven approaches proposed in recent years. It is to these that we now turn.

3.2 Planning-based approaches

In AI, the planning problem can be described as the process of identifying a sequence of one or more actions to satisfy a particular goal. An initial goal can be decomposed into sub-goals, satisfied by actions each of which has its preconditions and effects. In the classical planning paradigm (STRIPS; Fikes & Nilsson, 1971), actions are represented as tuples of such preconditions and effects.

The connection between planning and NLG lies in that text generation can be viewed as the execution of planned behaviour to achieve a communicative goal, where each action leads to a new state, that is, a change in a context that includes both the linguistic interaction or discourse history to date, but also the physical or situated context and the user’s beliefs and actions (see

Lemon, 2008; Rieser & Lemon, 2009; Dethlefs, 2014; Garoufi & Koller, 2013; Garoufi, 2014, for some recent perspectives on this topic). This perspective on NLG is therefore related to the view of ‘language as action’ (Clark, 1996), itself rooted in a philosophical tradition inaugurated by the work of Austin (1962) and Searle (1969). Indeed, some of the earliest AI work in this tradition (especially Cohen & Perrault, 1979; Cohen & Levesque, 1985) sought an explicit formulation of preconditions (akin to Searle’s felicity conditions) for speech acts and their consequences.

Given that there is in principle no restriction on what types of actions can be incorporated in a plan, it is possible for plan-based approaches to NLG to cut across the boundaries of many of the tasks that are normally encapsulated in the classic pipeline architecture, combining both tactical and strategic elements by viewing the problems of what to say and how to say it as part and parcel of the same set of operations. Indeed, there are important precedents in early work for a unified view of NLG as a hierarchy of goals, the KAMP system (Appelt, 1985) being among the best known examples. For instance, to generate referring expressions in KAMP, the starting point was reasoning about interlocutors’ beliefs and mutual knowledge, whereupon the system generated sub-goals that percolated all the way down to property choice and realisation, finally producing a referential NP whose predicted effect was to alter the hearer’s belief state about the referent (see Heeman & Hirst, 1995, for a similar approach to the generation of referring expressions in dialogue).

One problem with these perspectives, however, is that deep reasoning about beliefs, desires and intentions (or BDI, as it is often called following the work of Bratman, 1987) requires highly expressive formalisms and incurs considerable computational expense. One solution is to avoid general-purpose reasoning formalisms and instead adapt a linguistic framework to the planning paradigm for NLG.

3.2.1 Planning through the grammar

The idea of interpreting linguistic formalisms in planning terms is again prefigured in early NLG work. For example, some early systems (e.g. KPML, which we briefly discussed in the context of realisation in Section 2.6; Bateman, 1997) were based on Systemic-Functional Grammar (SFG; Halliday & Matthiessen, 2004), which can be seen as a precursor to contemporary planning-based approaches, since SFG models linguistic constructions as the outcome of a traversal through a decision network that extends backwards to pragmatic intentions. In a similar vein, both Hovy (1991) and Moore and Paris (1993) interpreted the relations of Rhetorical Structure Theory (Mann & Thompson, 1988) as operators for text planning.

Some recent approaches integrate much of the planning machinery into the grammar itself, viewing linguistic structures as planning operators. This requires grammar formalisms which integrate multiple levels of linguistic analysis, from pragmatics to morpho-syntax. It is common for contemporary planning-based approaches to NLG to be couched in the formalism of Lexicalised Tree

Adjoining Grammar (LTAG; Joshi & Schabes, 1997), though other formalisms, such as Combinatory Categorial Grammar (Steedman, 2000) have also been shown to be adequate to the task (see especially Nakatsu & White, 2010, for an approach to generation using Discourse Combinatory Categorial Grammar).

In an LTAG, pieces of linguistic structure (so-called elementary trees in a lexicon) can be coupled with semantic and pragmatic information that specify (a) what semantic preconditions need to obtain in order for the item to be felicitously used; and (b) what pragmatic goals the use of that particular item will achieve (see Stone & Webber, 1998; Garoufi & Koller, 2013; Koller & Striegnitz, 2002, for planning-based work using LTAG). As an example of how such a formalism could be deployed in a planning framework, let us focus on the task of referring to a target entity. Koller and Stone (2007) formulated the task in a way that obviates the need to distinguish between the content determination and realisation phases (an approach already taken by Stone & Webber, 1998). Furthermore, they do not separate sentence planning, REG and realisation, as is done in the traditional pipeline. Consider the sentence *Mary likes the white rabbit*. Simplifying the formalism for ease of presentation, we can represent the lexical item *likes* as follows (this example is based on Garoufi, 2014, albeit with some simplifications):

(12) **likes**(*u, x, y*) action:

PRECONDITIONS:

- The proposition that *x* likes *y* is part of the knowledge base (i.e. the statement is supported);
- *x* is animate;
- The current utterance *u* can be substituted into the derivation *S* under construction;

EFFECTS:

- *u* is now part of *S*
- New NP nodes for *x* in agent position and *y* in patient position have been set up (and need to be filled).

As in STRIPS, an operator consists of preconditions and effects. Note that the preconditions associated with the lexical item require support in the knowledge base (thus making reference to the input KB, which normally would not be accessible to the realiser), and include semantic information (such as that the agent needs to be animate). Having inserted *likes* as the sentence's main verb, we have two noun phrases which need to be filled by generating NPs for the arguments *x* and *y*. Rather than deferring this task to a separate REG module, Koller and Stone build referring expressions by associating further pragmatic preconditions on the linguistic operators (elementary trees) that will be incorporated in the referential NP. First, the entity must be part of the hearer's knowledge state, since an identifying description (say, to *x*) presupposes that

the hearer is familiar with it. Second, an effect of adding words to the NP (such as the predicates *rabbit* or *white*) is that the phrase excludes distractors, i.e. entities of which those properties are not true. In a scenario with one human being and two rabbits, only one of which (the y in our example) is white, the derivation would proceed by first updating the NP corresponding to y with *rabbit*, thereby excluding the human from the distractor set, but leaving the goal to distinguish y unsatisfied (since y is not the only rabbit). The addition of another predicate to the NP (*white*) does the trick.

A practical advantage to planning-based approaches is the availability of a significant number of off-the-shelf planners. Once the NLG task is formulated in an appropriate plan description language, such as the Planning Domain Definition Language (PDDL; McDermott, 2000), it becomes possible in principle to use any planner to generate text. However, planners remain beset by problems of efficiency. In a set of experiments on NLG tasks of differing complexity, Koller and Petrick (2011) noted that planners tend to spend significant amounts of time on preprocessing, though solutions could often be found efficiently once preprocessing was complete.

3.2.2 Stochastic planning under uncertainty using Reinforcement Learning

The approaches to planning we have discussed so far are largely rule-based and tend to view the relationship between a planned action and its consequences (that is, its impact on the context), as fixed (though exceptions exist, as in *contingency planning*, which generates multiple plans to address different possible outcomes; Steedman & Petrick, 2007).

As Rieser and Lemon (2009) note, this view is unrealistic. Consider a system that generates a restaurant recommendation. The consequences of its output (that is, the new state it gives rise to) are subject to noise arising from several sources of uncertainty. In part, this is due to trade-offs, for example, between needing to include the right amount of information while avoiding excessive prolixity. Another source of uncertainty is the user, whose actions may not be the ones predicted by the system. An instance of Meteer's (1991) generation gap can rear its head, for instance if a stochastic realiser renders the content of a message in an ambiguous, or excessively lengthy utterance (Rieser & Lemon, 2009), a problem that could be addressed by allowing different sub-tasks to share knowledge sources and be guided by overlapping constraints (Dethlefs & Cuayáhuitl, 2015, discussed below).

In short, planning a good solution to reach a communicative goal could be viewed as a stochastic optimisation problem (a theme we revisit in Section 3.3.3 below). This view is shared by many recent approaches based on Reinforcement Learning (RL; Lemon, 2008; Rieser & Lemon, 2009, 2011a), especially those that tackle NLG within a dialogue context. In this framework, generation can be modelled as a Markov decision process where states are associated with possible actions and each state-action pair is associated with a probability of moving from a state at time t to a new state at $t + 1$ via action a . Crucially for the

learning algorithm, transitions are associated with a reinforcement signal, via a reward function that quantifies the optimality of the generated output. Learning usually involves simulations in which different generation strategies or ‘policies’ – essentially, plans corresponding to possible paths through the state space – come to be associated with different rewards. The RL framework has been argued to be better at handling uncertainty in dynamic environments than supervised learning or classification, since these do not enable adaptation in a changing context (Rieser & Lemon, 2009). Rieser et al. (2011) showed that this approach is effective in optimising information presentation when generating restaurant recommendations. Janarthanam and Lemon (2014) used it to optimise the choice of information to select in a referring expression, given a user’s knowledge. The system learns to adapt its user model as the user acquires new knowledge in the course of a dialogue.

An important contribution of this work has been in exploring joint optimisation, where the policy learned satisfies multiple constraints arising from different sub-tasks of the generation process, by sharing knowledge across the sub-tasks. Lemon (2011) showed that joint optimisation can learn a policy that determines when to generate informative utterances or queries to seek more information from a user. Similarly, Cuayáhuitl and Dethlefs (2011) used hierarchical RL to jointly optimise the problem of finding and describing a short route description, while adapting to a user’s prior knowledge, giving rise to a strategy whereby the user is guided past landmarks that they are familiar with, while avoiding potentially confusing junctions. Also in a route-finding setting, Dethlefs and Cuayáhuitl (2015) develop a hierarchical model comprising a set of learning agents whose tasks range from content selection through realisation. They show that a joint framework in which agents share knowledge, outperforms an isolated learning framework in which each task is modelled separately. For example, the joint policy learns to give high-level navigation instructions, but switches to low-level instructions if the user goes off-track. Furthermore, utterances produced by the joint policy are less verbose and lead to shorter interactions overall.

In summary, NLG research within the planning paradigm has highlighted the desirability of developing unified formalisms to represent constraints on the generation process at multiple levels, whether this is done using AI-based planning formalisms (Koller & Petrick, 2011), or stochastically via Reinforcement Learning. Among its contributions, the latter line of work has shed light on the value of (a) hierarchical relationships among sub-problems; and (b) joint optimisation of different sub-tasks. Indeed, this work belongs to a much broader range of research on data-driven NLG, to which we turn our attention immediately below.

3.3 Data-driven approaches

Although the shift towards data-driven methods in NLG began somewhat later than in other areas of NLP, there is little doubt that this is now the dominant trend. In the remainder of this section, we start with an overview of methods

used to acquire training data for NLG – in particular, pairings of inputs (data) and outputs (text) – before turning to an overview of techniques and frameworks. One of the themes that will emerge from this overview is that, as in the case of planning, statistical methods often take a unified or ‘global’, rather than a modularised, view of the NLG process.

3.3.1 Acquiring data

As noted in Section 2, some NLG tasks support the transition to a stochastic approach fairly easily. For example, research on realisation often exploits the existence of treebanks from which input-output correspondences can be learned. Similarly, the emergence of corpora of referring expressions representing both input domains and output descriptions (e.g., Gatt et al., 2007; Viethen & Dale, 2011; Kazemzadeh et al., 2014; Gkatzia et al., 2015) has facilitated the development of probabilistic REG algorithms. Shared tasks have also contributed to the development of both data sources and methods (see Section 7). As we show in Section 4 below, recent work on image-to-text generation has also benefited from the availability of large datasets. For statistical, end-to-end generation in other domains, there is less of an embarrassment of riches. However, this situation is improving as methods to automatically align input data with output text are developed. Still, it is worth emphasising that many of these alignment approaches use data which is semi-structured, rather than the raw, numerical input (e.g. signals) used by the data-to-text systems that Reiter (2007), among others, drew attention to.

Currently, there are a number of data-text corpora in specific domains, notably weather forecasting (Reiter et al., 2005; Belz, 2008; Liang et al., 2009) and sports summaries (Barzilay & Lapata, 2005; Chen & Mooney, 2008). These usually consist of database records paired with free text. A promising recent trend is the introduction of statistical techniques that seek to automatically segment and align such data and text (e.g., Barzilay & Lapata, 2005; Liang et al., 2009; Konstas & Lapata, 2013). In an influential paper, Liang et al. (2009) described this framework in terms of a generative model that defines a distribution $p(w|s)$, for sequences of words w and input states s , with latent variables specifying the correspondence between w and s in terms of three main components: (i) the likelihood of database records being selected, given s ; (ii) the likelihood of certain fields being chosen for some record; (iii) the likelihood that a string of a certain length is generated given the records, fields and states. The parameters of the model can be found using the Expectation Maximization (EM) algorithm. An example alignment is shown in Figure 4.

These models perform alignment by identifying regular co-occurrences of segments of data and text. Koncel-Kedziorski and Hajishirzi (2014) go beyond this by proposing a model that exploits linguistic structure to align at varying resolutions. For example, (13) below is related to two observations in a soccer game log (an aerial pass and a miss), but can be further analysed into two sub-parts (indicated by indices 1 and 2 in our example), which individually map to these two sub-events.

Events:	<code>skyCover₁</code>	<code>temperature₁</code>			<code>windDir₁</code>		<code>windSpeed₁</code>
Fields:	<code>percent=0-25</code>	<code>time=6am-9pm</code>	<code>min=9</code>	<code>max=21</code>	<code>mode=S</code>	<code>wind</code>	<code>mean=20</code>
Text:	<code>cloudy ,</code>	<code>with</code>	<code>temperatures between</code>	<code>10</code>	<code>20 degrees .</code>	<code>around</code>	<code>20 mph .</code>

Figure 4: Database records aligned with text using minimal supervision. After Liang et al. (2009).

- (13) (Chamakh rises highest)₁ and (aims a header towards goal which is narrowly wide)₂.

A different approach to data acquisition is described by Mairesse and Young (2014), who use crowd-sourcing techniques to elicit realisations for semantic/pragmatic inputs describing dialogue acts in the restaurant domain (see Novikova & Rieser, 2016b, for another recent approach to crowd-sourcing in a different domain). The key to the success of this technique is the development of a semantics that is sufficiently transparent for use with non-specialists. In an earlier paper, Mairesse et al. (2010) describe a method to cut down on the amount of training data required for generation by using *uncertainty sampling* (Lewis & Catlett, 1994), whereby a system can be trained on a relatively small amount of input data; subsequently, the learned model is applied to new data, from which the system samples the cases of which it is least certain, forwarding these to a (possibly human) oracle for feedback, which potentially leads to a new training cycle.

While many of the stochastic end-to-end systems we discuss below rely on well-defined formalisms and typically need fairly precise alignments between inputs and portions of the output, more recent deep learning models (Section 3.3.5) have been based on partially aligned data (e.g. Wen et al., 2015; Lebret et al., 2016; Mei et al., 2016).

3.3.2 NLG based on language models

Given an alignment between data and text, one way of modelling the NLG process is to remain faithful to the division between strategic and tactical choices, using the statistical alignment to inform content selection, while deploying NLP techniques to acquire rules, templates or schemas (á la McKeown, 1985) to drive sentence planning and realisation.

Recall that the generative model of Liang et al. (2009) pairs data to text based on a sequential, Markov process, combining strategic choices (of DB records and fields) with tactical choices (of word sequences) into a single probabilistic model. In fact, Markov-based language modelling approaches continue to feature prominently in data-driven NLG. One of the earliest examples is Oh and Rudnicky (2002) in the context of a dialogue system in the travel domain,

where the input takes the form of a dialogue act (e.g. a query that the system needs to make to obtain information about the user’s travel plans) with the attributes to include (e.g. the departure city). Oh and Rudnicky’s approach encompasses both content planning and realisation. It relies on dialogue corpora annotated with utterance classes, that is, the type of dialogue act that each utterance is intended to fulfil. On this basis, they construct separate n -gram language models for each utterance class, as well as for word-classes that can appear in the input (for example, words corresponding to `departure city`). Content planning is handled by a model that predicts which attributes should be included in an utterance on the basis of recent dialogue history. Realisation is handled using a combination of templates and n -gram models. Thus, generation is conceived as a two-step (planning followed by realisation) process.

The reliance on standard language models has one potential drawback, in that such models are founded on a local history assumption, limiting the extent to which prior selections can influence current choices. An alternative, discriminative model (known to the NLP community at least since Ratnaparkhi, 1996) is logistic regression (Maximum Entropy). The foundations for this approach in NLG can be found in Ratnaparkhi (2000), who focussed primarily on realisation (albeit combined with elements of sentence planning). He compared two stochastic NLG systems based on a maximum entropy learning framework, to a baseline NLG system. The first of these (NLG2 in Ratnaparkhi’s paper) uses a conditional language model that generates sentences in an incremental, left-to-right fashion, by predicting the best word given both the preceding history (as in standard n -gram models) and the semantic attributes that remain to be expressed. The second (NLG3) augments the model with syntactic dependency relations, performing generation by recursively predicting the left and right children of a given constituent. In an evaluation based on judgements of correctness, Ratnaparkhi found that the system augmented with dependencies was generally preferred.

In later work, Angeli et al. (2010) describe an approach to end-to-end NLG that maintains a separation between content selection, sentence planning and realisation, modelling each process as a sequence of decisions in a log-linear framework, where choices can be conditioned on arbitrarily long histories of previous decisions. This enables them to handle long-range dependencies, such as coherence relations, more flexibly (e.g., a model can incorporate the information that a weather report which describes wind speed should do so after mentioning wind direction; see Barzilay & Lapata, 2005, for similar insights based on global optimisation). The separation of tasks is maintained insofar as a different set of features can be used to inform decisions at each stage of the process. Sentence planning and realisation decisions are based on templates acquired from corpus texts: a template is selected based on its likelihood given the database fields selected during content selection.

Mairesse and Young (2014) describe a different approach, which also relies on alignments between database records and text, and seeks a global solution to generation, without a crisp distinction between strategic and tactical components. In this case, the basic representational framework is a tree of the sort

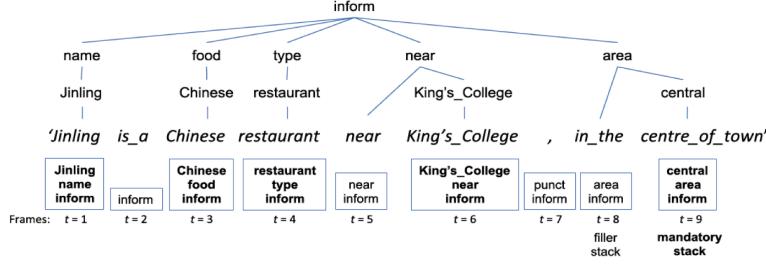


Figure 5: Tree structure for a dialogue act, after Mairesse and Young (2014). Leaves correspond to word sequences. Non-terminal nodes are semantic attributes, shown at the bottom as semantic stacks. Stacks in bold represent mandatory content.

shown in Figure 5. The root indicates a dialogue act type (in the example, the dialogue act seeks to **inform**). Leaves in the tree correspond to words or word sequences, while nonterminals are *semantic stacks*, that is, the pieces of input to which the words correspond. In this framework, content selection and realisation can be solved jointly by searching for the optimal stack sequence for a given dialogue act, and the optimal word sequence corresponding to that stack sequence. Mairesse and Young use a factored language model (FLM), which extends n-gram models by conditioning probabilities on different utterance contexts, rather than simply on word histories. Given an input dialogue act, generation works by applying a Viterbi search through the FLM at each of the following stages: (a) mandatory semantic stacks are identified for the dialogue act; (b) these are enriched with possible non-mandatory stacks (those which are not in boldface in Figure 5), usually corresponding to function words; (c) realisations are found for the stack sequence. The approach is also extended to deal with n -best realisations, as well as to handle variation, in the form of paraphrases for the same input.

3.3.3 NLG as classification and optimisation

An alternative way to think about NLG decisions at different levels is in terms of classification, already encountered in the context of specific tasks, such as content determination (e.g., Duboue & McKeown, 2003) and realisation (e.g., Filippova & Strube, 2007). Since generation is ultimately about choice-making at multiple levels, one way to model the process is by using a cascade of classifiers, where the output is constructed incrementally, so that any classifier C_i uses as (part of) its input the output of a previous classifier C_{i-i} . Within this framework, it is still possible to conceive of NLG in terms of a pipeline. As Marciniak and Strube (2005) note, an alternative way of thinking about it is

in terms of a weighted, multi-layered lattice, where generation amounts to a best-first traversal: at any stage i , classifier C_i produces the most likely output, which leads to the next stage C_{i+1} along the most probable path. This generalisation is conceptually related to the view of NLG in terms of policies in the Reinforcement Learning framework (see Section 3.2.2 above), which define a traversal through sequences of states which may be hierarchically organised (as in the work of Dethlefs & Cuayahuitl, 2015, for example).

Marciniak and Strube (2004) start from a small corpus of manually annotated texts of route descriptions, dividing generation into a series of eight classification problems, from determining the linear precedence of discourse units, to determining the lexical form of verbs and the type of their arguments. Generation decisions are taken using the instance-based KStar algorithm, which is shown to outperform a majority baseline on all classification decisions.⁵

A similar framework was recently adopted by Zarrieß and Kuhn (2013), once again taking as their starting point textual data annotated with a dependency representation, as shown in (3.3.3) below, where referents are marked v and p and the implicit head of the dependency is underlined.

- (14) *Junge Familie_{v:0} auf dem Heimweg_{poss:v} ausgeraubt_{ag:p}*
 Young family on the way home robbed
 ‘A young family was robbed on their way home.’

These authors use a sequence of classifiers to perform referring expression generation and realisation. They use a ranking model based on Support Vector Machines which, given an input dependency representation extracted from annotated text such as (3.3.3), performs two tasks in either order: (a) mapping the input to a shallow syntactic tree for linearisation; and (b) inserting referring expressions. Interestingly, Zarrieß and Kuhn (2013) observe that the performance of either task is order-dependent, in that both classification tasks perform worse when they are second in the sequence. They observe a marginal improvement when the tasks are performed in parallel, but achieve the best performance in a *revision-based* architecture, where syntactic mapping is followed by referring expression insertion, followed by a revision of the syntax.

Classification cascades for NLG maintain a clean separation between tasks, but research in this area has echoed earlier concerns about pipelines in general (see Section 3.1), the main problem being error propagation. Infelicitous choices will of course impact classification further downstream, a situation analogous to the problem of the generation gap. The conclusion by Zarrieß and Kuhn (2013) in favour of a revision-based architecture, brings our account full circle, in that a well-known solution is shown to yield improvements in a new framework.

Our discussion so far has repeatedly highlighted the fact that a sequential organisation of NLG tasks is susceptible to error propagation, whether this takes the form of classifier errors, or decisions in a rule-based module that have a

⁵Instance-based approaches to NLG are also discussed by Varges and Mellish (2010), albeit in an overgenerate-and-rank approach where rules overgenerate candidates, which are then ranked by comparison to the instance base.

negative impact on downstream components. A potential solution is to view generation as an optimisation problem, where the best combination of decisions is sought in an exponentially large space of possible combinations. We have encountered the use of optimisation techniques, such as Integer Linear Programming (ILP) in the context of aggregation and content determination (Section 2.3). For example, Barzilay and Lapata (2006) group content units based on their pairwise similarity, with an optimisation step to identify a set of pairs that are maximally similar. ILP has also been exploited by Marciniak and Strube (2005), as a means to counteract the error propagation problem in their original classification-based approach (Marciniak & Strube, 2004). Conceptually, the optimisation framework is simple:

1. Each NLG task is once again modelled as classification or label-assignment, but this time, labels are modelled as binary choices (either a label is assigned or not), associated with a cost function, defined in terms of the probability of a label in the training data;
2. Pairs of tasks which are strongly inter-dependent (for example, syntactic choices and REG realisations, in the example from Zarrieß & Kuhn, 2013) have a cost based on the joint probability of their labels;
3. An ILP model seeks the global labelling solution that minimises the overall cost, with the added constraint that if one of a pair of correlated labels $\langle l_i, l_j \rangle$ is selected, the other must be too.

This optimisation solution has been shown to outperform different versions of the classification pipeline originally proposed by Marciniak and Strube (2004), much as the results of Dethlefs and Cuayahuitl (2015), discussed above, showed that reinforcement learning of a joint policy produces better dialogue interactions than learning isolated policies for separate NLG tasks. Another advantage of this framework is that for planning-based approaches, it relies on a formalism for which off-the-shelf tools are available. As with planning, however, efficiency may be a challenge, given the large space of possibilities for an optimisation problem.

3.3.4 NLG as ‘parsing’

In recent years, there has been a resurgence of interest in viewing generation in terms of probabilistic context-free grammar (sc cfg) formalisms, or even as the ‘inverse’ of semantic parsing. For example, Belz (2008) formalises the NLG problem entirely in terms of CFGs: a base generator expands inputs (bits of weather data in this case) by applying CFG rules; corpus-derived probabilities are then used to control the choice of which rules to expand at each stage of the process. The base generator in this work is hand-crafted. However, it is possible to extract rules or templates from corpora, as has been done for aggregation rules (Stent & Molina, 2009; White & Howcroft, 2015, and Section 2.3), and also for more general statistical approaches to sentence planning and realisation

in a text-to-text framework (e.g., Kondadadi et al., 2013). Similarly, approaches to NLG from structured knowledge bases, expressed in formalisms such as RDF, have described techniques to extract lexicalised grammars or templates from such inputs paired with textual descriptions (Ell & Harth, 2014; Duma & Klein, 2013; Gyawali & Gardent, 2014).

The work of Mooney and colleagues (Wong & Mooney, 2007; Chen & Mooney, 2008; Kim & Mooney, 2010) has compared a number of different generation strategies inspired by the WASP semantic parser (Wong & Mooney, 2007), which uses probabilistic synchronous CFG rules learned from pairs of utterances and their semantic representations using statistical machine translation techniques. Chen and Mooney (2008) use this framework for generation both by adapting WASP in a generation framework, and by further adapting it to produce a new system, WASPER-GEN. While WASP seeks to maximise the probability of a meaning representation (MR) given a sentence, WASPER-GEN does the opposite, seeking the maximally probable sentence given an input MR, as it were, learning a translation model from meaning to text. When trained on a dataset of sportscasts (the ROBOCUP dataset), WASPER-GEN outperforms WASP on corpus-based evaluation metrics, and is shown to achieve a level of fluency and semantic correctness which approaches that of human text, based on subjective judgements by experimental participants. Note, however, that this framework focusses mainly on tactical generation. Content determination is performed separately, using a variant of the EM-algorithm to converge on a probabilistic model that predicts which events or predicates should be mentioned.

By contrast, the work of Konstas and Lapata (Konstas & Lapata, 2012, 2013), which also relies on CFGs, uses a unified framework throughout. The starting point is an alignment of text with database records, extending the proposal by Liang et al. (2009). The process of converting input data to output text is modelled in terms of rules which implicitly incorporate different types of decisions. For example, given a database of weather records, the rules might take the simplified form shown below,

- (15) $R(windSpeed) \rightarrow FS(temperature), R(rain)$
- (16) $FS(windSpeed, min) \rightarrow FS(windSpeed, max)FS(windSpeed, max)$
- (17) $FS(windSpeed, min) \rightarrow W(windSpeed, min)$

where R stands for a database record, FS is a set of fields, W is a word sequence, and all rules have associated probabilities that condition the RHS on the LHS, akin to the PCFGs used in parsing. These rules specify that a description of *windSpeed* (15) should be followed in the text by a temperature and a rain report. According to rule (16), minimum windspeed should be followed by maximum windspeed with a certain probability, while rule (17) expands the minimum windspeed rule to a sequence of words according to a bigram language model (Konstas & Lapata, 2012). Konstas and Lapata (2012) pack the set of rules acquired from the alignment stage into a hypergraph, and treat generation as decoding to find the maximally likely word sequence.

Under this view, generation is akin to inverted parsing. Decoding proceeds using an adaptation of the CYK algorithm. Since the model defining the mapping from input to output does not incorporate fluency heuristics, the decoder is interleaved with two further sources of linguistic knowledge by Konstas and Lapata (2013): (a) a weighted finite-state automaton (representing an n-gram language model); and (b) a dependency model (cf. Ratnaparkhi, 2000, , also discussed above).

3.3.5 Deep learning methods

We conclude our discussion of statistical methods with an overview of applications of deep neural network (NN) architectures to generation and related tasks. The decision to dedicate a separate section is warranted by the recent, renewed interest in these models, as well as the comparatively small number of studies that have adopted this framework in NLG to date (with the exception of caption generation from images, which we review in Section 4) . However, this is undoubtedly one of the biggest growth areas in current NLG research, as in NLP more generally (see Goldberg, 2016, for an NLP-focussed overview).

The recent resurgence of interest in NNs is in part due to advances in hardware that can support resource-intensive learning problems (Goodfellow et al., 2016). More importantly, NNs are designed to learn representations at increasing levels of abstraction by exploiting backpropagation (LeCun et al., 2015; Goodfellow et al., 2016). Such representations are dense, low-dimensional, and distributed, making them especially well-suited to capturing grammatical and semantic generalisations (see Mikolov et al., 2013; Luong et al., 2013; Pennington et al., 2014, *inter alia*). NNs have also scored notable successes in sequential modelling using feedforward networks (Bengio et al., 2003; Schwenk & Gauvain, 2005), log-bilinear models (Mnih & Hinton, 2007) and recurrent neural networks (RNNs Mikolov et al., 2010), including RNNs with long short-term memory units (Zaremba et al., 2015). Their main advantage over standard n -gram models is that they represent sequences of varying lengths, while avoiding both data sparseness and an explosion in the number of parameters through the projection of histories into a low-dimensional space, so that similar histories have joint representations.

Long short-term memory architectures (LSTM) are a further development of RNNs equipped with memory cells and multiplicative gates that control how information is retained or forgotten. This also enables them to handle long-range dependencies. Ultimately, the goal of such models is to learn a conditional probability $p(Y|T)$ between an output sequence Y and an input sequence T whose length may differ from that of Y (Sutskever et al., 2014; LeCun et al., 2015). Their ability to map from variable-length inputs to fixed-dimensional vector representations, conditioning the probability of the next element in an output sequence on the fixed-dimensional vector and the previously emitted tokens of the sequence, make them well-suited to tasks such as Machine Translation (MT; e.g. Kalchbrenner & Blunsom, 2013; Bahdanau et al., 2015), which can be viewed as instances of more general sequence-to-sequence (aka *seq2seq*)

tasks. Sutskever et al. (2014) showed that LSTM architectures can be adapted to different *seq2seq* problems by decoupling them into an initial LSTM encoder for the input and a subsequent decoder. This has given impetus to research on multitask learning, where different tasks share parts of a network architecture and the corresponding representations (Dong et al., 2015; Luong et al., 2016). For example, Luong et al. (2016) showed that improvements can be obtained in MT with a SEQ2SEQ architecture whose encoder is co-trained on a parsing dataset; similarly, decoding improved when the decoder (for the target language English) was shared with a caption generator.

As a matter of fact, applications of NNS in NLG hark back at least to Kukich (1987), though her work was tentative and restricted to small-scale examples. Since the early 1990s, when interest in neural approaches waned in the NLP and AI communities, cognitive science research has continued to explore their application to syntax and language production (e.g., Elman, 1990, 1993; Chang et al., 2006).

In the past few years, research on neural models for generation has begun to take off. A starting point can be found in the work of Sutskever et al. (2011), who showed that a character-level LSTM RNN could be used to generate grammatical English sentences. Since then, a number of NLG applications have appeared. For instance, Zhang and Lapata (2014a) focus on poetry generation in Chinese using RNNS, where verses are generated by predicting characters based on (a) the previous verses in the poem, represented by a convolutional sentence model (Kalchbrenner & Blunsom, 2013); and (b) the previous characters in the current verse. In dialogue, both Wen et al. (2015) and Serban et al. (2016) use RNNS to predict the next utterance in a dialogue context, while (Goyal et al., 2016) show that the generation of dialogue acts improves when modelled using a character-based, rather than a word-based RNN.

There has also been some work applying NNS to data-to-text generation (Mei et al., 2016; Lebret et al., 2016). The LSTM-based architecture proposed by Mei et al. (2016) is based on the encoder-decoder framework, with a design that broadly reflects a division into content selection and realisation. The application domain is WEATHERGOV data (Angeli et al., 2010). The starting point is a bidirectional LSTM RNN encoder which maps input records to a hidden state, followed by an aligner which models content selection. The aligner, inspired by the MT work of Bahdanau et al. (2015) and the attention-based image captioning work of Xu et al. (2015), determines which records to mention as a function of their prior probability and the likelihood of their alignment with words in the vocabulary; a further refinement step weights the outcomes of the alignment with the priors, making it more likely that more important records will be verbalised. Finally, a decoder RNN outputs a word-by-word sequence, at each time step computing a probability distribution over words given the previously generated context and the records. LSTMs enable the handling of long-range dependencies between records and descriptors, which the log-linear model of Angeli et al. (2010) factored in explicitly (see Section 3.3.2 above).

The work of Lebret et al. (2016), by contrast, restricts generation to the initial sentence of wikipedia biographies from the corresponding wiki fact table

and models content selection and realisation jointly in a feedforward NN (Bengio et al., 2003), conditioning output word probabilities on both local context and global features obtained from the input table. This biases the model towards full coverage of the contents of a field. For example, a field in the table containing a person’s name typically consists of more than one word and the model should concatenate the words making up the entire name. This simpler model can therefore also be characterised as incorporating an attentional mechanism.

3.4 Discussion

An important theme that has emerged from recent work is the blurring of boundaries between tasks that are encapsulated in traditional architectures. This is evident in planning-based approaches, but perhaps the most radical break from this perspective arises in stochastic data-to-text systems which capitalise on alignments between input data and output text, combining content-oriented and linguistic choices within a unified framework. Among the open questions raised by research on stochastic NLG is the extent to which sub-tasks need to be jointly optimised and, if so, which knowledge sources should be shared among them.

An outstanding issue is the balancing act between achieving adequate textual output versus doing so efficiently and robustly. Early approaches that departed from a pipeline architecture tended to sacrifice the latter in favour of the former; this was the case in revision-based and blackboard architectures. The same is to some extent true of planning-based approaches which are rooted in paradigms with a long history in AI: As recent empirical work has shown (Koller & Petrick, 2011), these too are susceptible to considerable computational cost, though this comes with the advantage of a unified view of language generation that is also compatible with well-understood linguistic formalisms, such as LTAG. Stochastic approaches present a different problem, namely, that of acquiring the right data to construct the necessary statistical models. While such data is or can be made available, for tasks such as recommendations, brief weather reports, or sports summaries, it remains to be seen whether existing techniques for data-text alignment can be scaled up to domains where large volumes of heterogeneous data (numbers, symbols etc) are the norm, and where longer texts need to be generated.

In any case, one important outcome of much recent data-driven research in NLG is the emphasis on unified formalisms – from CFGs to markov processes – that underlie the text generation process at every level. Another interesting development is the use of crowd-sourcing techniques to produce data that aligns non-linguistic input representations with text (Mairesse & Young, 2014; Novikova & Rieser, 2016b).

As deep learning approaches become more popular – and, as we shall see in the next section, they are now the dominant approach in certain tasks, such as generating image captions – the need for precise alignment could become less acute, as looser input-output couplings can constitute adequate training data (e.g. Wen et al., 2015). As these techniques become better understood, they

are likely to feature more heavily in a broader range of NLG tasks, as well as end-to-end NLG systems. As a recent opinion piece put it (Manning, 2015), perhaps their attraction for NLP, apart from the advantages of using distributed representations, lies in that they encourage the practitioner to focus on design, that is, on how an architecture can handle the different sub-parts of a complex problem, perhaps along the lines detailed in Section 2 above. Might this entail a renewed emphasis on modular, multi-levelled approaches to NLG, with complex architectures whose components deal with different tasks? On the other hand, research on multi-task learning in the *seq2seq* paradigm may also open up new possibilities for learning how to solve multiple NLG sub-tasks within a single framework, as many approaches discussed in the previous sub-sections seek to do.

In the following sections, we turn our attention away from standard tasks and the way they are organised, focussing on three broad topics – image-to-text generation, stylistic variation and computational creativity – in which NLG research has also intersected with research in other areas.

4 The vision-language interface: Image captioning and beyond

Over the past few years, there has been an explosion of interest in the task of automatically generating captions for images, as part of a broader endeavour to investigate the interface between vision and language (Barnard, 2016). Image captioning is arguably a paradigm case of data-to-text generation, where the input comes in the form of an image. The task has become a research focus not only in the NLG community but also in the computer vision community, raising the possibility of more effective synergies between the two groups of researchers. Apart from its practical applications, the grounding of language in perceptual data has long been a matter of scientific interest in AI (see Winograd, 1972; Harnad, 1990; Roy & Reiter, 2005, for a variety of theoretical views on the computational challenges of the perception-language interface).

Figure 6 shows some examples of caption generation, sampled from publications spanning approximately 6 years. Current caption generation research focusses mainly on what Hodosh et al. (2013) refer to as *concrete conceptual* image descriptions of elements directly depicted in a scene. As Donahue et al. (2015) put it, image captioning is a task whose input is static and non-sequential (an image, rather than, say, a video), whereas the output is sequential (a multi-word text), in contrast to non-sequential outputs such as object labels (e.g. Duygulu et al., 2002; Ordonez et al., 2016, among others).

Our discussion will be brief, since image captioning has recently been the subject of an extensive review by Bernardi et al. (2016), and has also been discussed against the background of broader issues in research on the vision-language interface by Barnard (2016). While the present section draws upon these sources, it is organised in a somewhat different manner, also bringing out



Figure 6: Some caption generation examples

the connections with NLG more explicitly.

4.1 Data

A detailed overview of datasets is provided by Bernardi et al. (2016), while Ferraro et al. (2015) offer a systematic comparison of datasets for both caption generation and visual question answering with an accompanying online resource⁶.

Datasets typically consist of images paired with one or more human-authored captions (mostly in English) and vary from artificially created scenes (Zitnick et al., 2013) to real photographs. Among the latter, the most widely used are Flickr8k (Hodosh et al., 2013), Flickr30k (Young et al., 2014) and MS-COCO (Lin

⁶<http://visionandlanguage.net>

et al., 2014). Datasets such as the SBU1M Captioned Photo Dataset (Ordonez et al., 2011) include naturally-occurring captions of user-shared photographs on sites such as Flickr; hence the captions included therein are not restricted to the concrete conceptual. There are also a number of specialised, domain-specific datasets, such as the Caltech UCSD Birds dataset (CUB; Wah et al., 2011).

There have also been a number of shared tasks in this area, including the COCO ('Common Objects in Context') Captioning Challenge⁷, organised as part of the Large-Scale Scene Understanding Challenge (LSUN)⁸ and the Multimodal Machine Translation Task (Elliott et al., 2016). We defer discussion of evaluation of image captioning systems to Section 7 of this paper, where it is discussed in the context of NLG evaluation as a whole.

4.2 The core tasks

There are two logically distinguishable sub-tasks in an image captioning system, namely, image analysis and text generation. This is not to say that they need to be organised separately or sequentially. However, prior to discussing architectures as such, it is worth briefly giving an overview of the methods used to deal with these two tasks.

4.2.1 Image analysis

There are three main groups of approaches to treating visual information for captioning purposes.

Detection Some systems rely on computer vision methods for the detection and labelling of objects, attributes, ‘stuff’ (typically mapped to mass nouns, such as *grass*), spatial relations, and possibly also action and pose information. This is usually followed by a step mapping these outputs to linguistic structures ('sentence plans' of the sort discussed in Section 2 and 3), such as trees or templates (e.g. Kulkarni et al., 2011; Yang et al., 2011; Mitchell et al., 2012; Elliott & De Vries, 2015; Yatskar et al., 2014; Kuznetsova et al., 2014). Since performance depends on the coverage and accuracy of detectors (Kuznetsova et al., 2014; Bernardi et al., 2016), some work has also explored generation from gold standard image annotations (Elliott & Keller, 2013; Wang & Gaizauskas, 2015; Muscat & Belz, 2015) or artificially created scenes in which the components are known in advance (Ortiz et al., 2015).

Holistic scene analysis Here, a more holistic characterisation of a scene is used, relying on features that do not typically identify objects, attributes and the like. Such features include RGB histograms, scale-invariant feature transforms (SIFT; Lowe, 2004), or low-dimensional representations of spatial structure (as in GIST; Oliva & Torralba, 2001), among others. This kind of image processing

⁷<http://mscoco.org/dataset/#captions-challenge2015>

⁸<http://lsun.cs.princeton.edu/2016/>

is often used by systems that frame the task in terms of retrieval, rather than caption generation proper. Such systems either use a unimodal space to compare a query image to training images before caption retrieval (e.g. Ordonez et al., 2011; Gupta et al., 2012), or exploit a multimodal space representing proximity between images and captions (e.g. Hodosh et al., 2013; Socher et al., 2014).

Dense image feature vectors Given the success of convolutional neural networks (CNN) for computer vision tasks (cf. e.g., LeCun et al., 2015), many deep learning approaches use features from a pre-trained CNN such as AlexNet (Krizhevsky et al., 2012), VGG (Simonyan & Zisserman, 2015) or Caffe (Jia et al., 2014). Most commonly, caption generators use an activation layer from the pre-trained network as their input features (e.g. Kiros et al., 2014; Karpathy et al., 2014; Karpathy & Fei-Fei, 2015; Vinyals et al., 2015; Mao et al., 2015a; Xu et al., 2015; Yagcioglu et al., 2015; Hendricks et al., 2016b).

4.2.2 Text generation or retrieval

Depending on the type of image analysis technique, captions can be generated using a variety of different methods, of which the following are well-established.

Using templates or trees Systems relying on detectors can map the output to linguistic structures in a sentence planning stage. For example, objects can be mapped to nouns, spatial relations to prepositions, and so on. Yao et al. (2010) use semi-supervised methods to parse images into graphs and then generate text via a simple grammar. Other approaches rely on sequence classification algorithms, such as Hidden Markov Models (Yang et al., 2011) and conditional random fields (Kulkarni et al., 2011, 2013). Kulkarni et al. (2013, see the example in Figure 6b) experiment with both templates and web-derived n -gram language models, finding that the former are more fluent, but suffer from lack of variation, an issue we also addressed earlier, in connection with realisation (Section 2.6).

In the Midge system (Mitchell et al., 2012, see Figure 6d for an example caption), input images are represented as triples consisting of object/stuff detections, action/pose detections and spatial relations. These are subsequently mapped to $\langle \text{noun}, \text{verb}, \text{preposition} \rangle$ triples and realised using a tree substitution grammar. This is further enhanced with the ability to ‘hallucinate’ likely words using a probabilistic model, that is, insert words which are not directly grounded in the detections performed on the image itself, but have a high probability of occurring, based on corpus data. In a human evaluation, Midge was shown to outperform both the system by (Kulkarni et al., 2011) and (Yang et al., 2011) on a number of criteria, including humanlikeness and correctness.

Elliott and Keller (2013) use *visual dependency representations* (VDR), a dependency grammar-like formalism to describe spatial relations between objects based on physical features such as proximity and relative position. Detections from an image are mapped to their corresponding VDR relations prior to generation (see also Elliott & De Vries, 2015, and the example in Figure 6c). Ortiz

et al. (2015) use ILP to identify pairs of objects in abstract scenes (Zitnick & Parikh, 2013) before mapping them to a VDR. Realisation is framed as a machine translation task over VDR-text pairs. A similar concern with identifying spatial relations is found in the work of Lin and Kong (2015), who use scene graphs as input to a grammar-based realiser. Muscat and Belz (2015) propose a naive Bayes model to predict spatial prepositions based on image features such as object proximity and overlap.

Using language models Using language models has the potential advantage of facilitating joint training from image-language pairs. It may also yield more expressive or creative captions if it is used to overcome the limitations of grammars or templates (as shown by the example of Midge; Mitchell et al., 2012). In some cases, n-gram models are trained on out-of-domain data, the approach taken by (Li et al., 2011, using web-scale n -grams) and Fang et al. (2015, using a maximum entropy language model). Most deep learning architectures use language models in the form of vanilla RNNs or long short-term memory networks (e.g. Kiros et al., 2014; Vinyals et al., 2015; Donahue et al., 2015; Karpathy & Fei-Fei, 2015; Xu et al., 2015; Hendricks et al., 2016b, 2016a; Mao et al., 2016). These architectures model caption generation as a process of predicting the next word in a sequence. Predictions are biased both by the caption history generated so far (or the start symbol for initial words) and by the image features which, as noted above, are typically features extracted from a CNN trained on the object detection task.

Caption retrieval and recombination Rather than generate captions, some systems retrieve them based on training data. The advantage of this is that it guarantees fluency, especially if retrieval is of whole, rather than partial, captions. Hodosh et al. (2013) used a multimodal space to represent training images and captions, framing retrieval as a process of identifying the nearest caption to a query image. The idea of ‘wholesale’ caption retrieval has a number of precedents. For example Farhadi et al. (2010) use Markov random fields to parse images into $\langle object, action, scene \rangle$ triples, paired with parsed captions. A caption for a query image is retrieved by comparing it to the parsed images in the training data, finding the most similar based on WordNet. Similarly, the Im2Text (Ordonez et al., 2011) system ranks candidate captions for a query image. Devlin et al. (2015b) use a k nearest neighbours approach, with caption similarity quantified using BLEU (Papineni et al., 2002) and CIDEr (Vedantam et al., 2015). A different view of retrieval is proposed by Feng and Lapata (2010), who use extractive summarisation techniques to retrieve descriptions of images and associated narrative fragments from their surrounding text in news articles.

A potential drawback of wholesale retrieval is that captions in the training data may not be well-matched to a query image. For instance, Devlin et al. (2015b) note that the less similar a query is to training images, the more generic the caption returned by the system. A possible solution is to use partial matches,

retrieving and recombining caption fragments. Kuznetsova et al. (2014) use detectors to match query images to training instances, retrieving captions in the form of parse tree fragments which are then recombined. Mason and Charniak (2014) use a domain-specific dataset to extract descriptions and adapt them to a query image using a joint visual and textual bag-of-words model. In the deep learning paradigm, both Socher et al. (2014) and Karpathy et al. (2014) use word embeddings derived from dependency parses, which are projected, together with CNN image features, into a multimodal space. Subsequent work by Karpathy and Fei-Fei (2015) showed that this fine-grained pairing works equally well with word sequences, eschewing the need for dependency parsing.

Recently, Devlin et al. (2015a) compared nearest-neighbour retrieval approaches to different types of language models for caption generation, specifically, the Maximum Entropy approach of Fang et al. (2015), an LSTM-based approach and RNNs which are coupled with a CNN for image analysis (e.g. Vinyals et al., 2015; Donahue et al., 2015; Karpathy & Fei-Fei, 2015). A comparison of the linguistic quality of captions suggested that there was a significant tendency for all models to reproduce captions observed in the training set, repeating them for different images in the test set. This could be due to a lack of diversity in the data, which might also explain why the nearest neighbour approach compares favourably with language model-based approaches.

4.3 How is language grounded in visual data?

As the foregoing discussion suggests, views on the relationship between visual and linguistic data depend on how each of the two sub-tasks is dealt with. Thus, systems which rely on detections tend to make a fairly clear-cut distinction between input processing and content selection on the one hand, and sentence planning and realisation on the other (e.g. Kulkarni et al., 2011; Mitchell et al., 2012; Elliott & Keller, 2013). The link between linguistic expressions and visual features is mediated by the outcomes of the detectors. For example, Midge (Mitchell et al., 2012) uses the object detections to determine which nouns to mention, before fleshing out the caption with attributes (mapped to adjectives) and verbs. Similarly, Elliott and Keller (2013) uses VDRs to determine spatial expressions.

Retrieval-based systems relying on unimodal or multimodal similarity spaces represent the link between linguistic expressions and image features more indirectly. Here, similarity plays the dominant role. In a unimodal space (Ordonez et al., 2011; Gupta et al., 2012; Mason & Charniak, 2014; Kuznetsova et al., 2012, 2014), it is images which are compared, with (partial) captions retrieved based on image similarity. A number of deep learning approaches also broadly conform to this scheme. For example, both Yagcioglu et al. (2015) and (Devlin et al., 2015b) retrieve and rank captions for a query image, using a CNN for the representation of the visual space. By contrast, multimodal spaces involve a direct mapping between visual and linguistic features (e.g. Hodosh et al., 2013; Socher et al., 2014; Karpathy et al., 2014), enabling systems to map from images to ‘similar’ – that is, related or relevant – captions.

Much interesting work on vision-language integration is being carried out with deep learning models. Kiros et al. (2014) introduced multimodal neural language models (MRNN), experimenting with two main architectures. Their Modality-Biased Log-Bilinear Model (MLBL-B) uses an additive bias to predict the next word in a sequence based on both the linguistic context and CNN image features. The Factored 3-way Log-Bilinear Model (MLBL-F) also gates the representation matrix for a word with image features. In a related vein, Donahue et al. (2015) propose a combined CNN + LSTM architecture (also used by Venugopalan et al., 2015b, 2015a, for video captioning) where the next word is predicted as a function of both previous words and image features. In one version of the architecture, they inject CNN features into the LSTM at each time-step. In a second version, they use two stacked LSTMs, the first of which takes CNN features and produces an output which constitutes the input to the next LSTM to predict the word. Finally, Mao et al. (2015a) experiment with various MRNN configurations, obtaining their best results with an architecture in which there are two word embedding layers preceding the recurrent layer, which is in turn projected into a multimodal layer where linguistic features are combined with CNN features. An example caption is shown in Figure 6e above.

These neural network models shed light on the consequences of combining the two modalities at different stages, reflecting the point made by Manning (2015, cf. Section 3.3.5) that this paradigm encourages a focus on architectures and design. In particular, image features can be used to bias the recurrent, language generation layer – at the start, or at each time-step of the RNN – as in the wrk of Donahue et al. (2015). Alternatively, the image features can be combined with linguistic features at a stage following the RNN, as in the work of Mao et al. (2015a).

4.4 Vision and language: Current and future directions for NLG

Image to text generation is one area of NLG where there is a clear dominance of deep learning methods. Current work focusses on a number of themes:

1. Generalising beyond training data is still a challenge, as shown by the work of Devlin et al. (2015a). More generally, dealing with novel images remains difficult, though experiments have been performed on using out-of-domain training data to expand vocabulary (Ordonez et al., 2013), learn novel concepts (Mao et al., 2015b) or transfer features from image regions containing known labels, to similar, but previously unattested ones (Hendricks et al., 2016b, from which an example caption is shown in Figure 6f). Progress in zero-shot learning, where the aim is to identify or categorise images for which little or no training data is available, is likely to contribute to the resolution of data sparseness problems (e.g. Antol et al., 2014; Elhoseiny et al., 2017).
2. Attention is also being paid to what Barnard (2016) refers to as *localisation*, that is, the association of linguistic expressions with parts of images,

and the ability to generate descriptions of specific image regions. Recent work includes that of Karpathy and Fei-Fei (2015), Johnson et al. (2016) and (Mao et al., 2016), who focus on unambiguous descriptions of specific image regions and/or objects in images (see Section 2.5 above for some related work). Attention-based models are a further development on this front. These have been exploited in various SEQ2SEQ tasks, notably for machine translation (Bahdanau et al., 2015). In the case of image captioning, the idea is to allocate variable weights to portions of captions in the training data, depending on the current context, to reflect the ‘relevance’ of a word given previous words and an image region (Xu et al., 2015).

3. Recent work has also begun to explore generation from images that goes beyond the concrete conceptual, for instance, producing explanatory descriptions (Hendricks et al., 2016a). A further development is work on Visual Question Answering, where rather than descriptive captions, the aim is to produce responses to specific questions about images (Geman et al., 2015; Barnard, 2016; Antol et al., 2015; Malinowski et al., 2016). Recently, a new dataset was proposed providing both concrete conceptual and ‘narrative’ texts coupled with images (Huang et al., 2016), a promising new direction for this branch of NLG.
4. There is a growing body of work that generalises the task from static inputs to sequential ones, especially videos (e.g. Kojima et al., 2002; Regneri et al., 2013; Venugopalan et al., 2015b, 2015a). Here, the challenges include handling temporal dependencies between scenes, but also dealing with redundancy.

5 Variation: Generating text with style, personality and affect

Based on the preceding sections, the reader could be excused for thinking that NLG is mostly concerned with delivering factual information, whether this is in the form of a summary of weather data, or a description of an image. This bias was also flagged in the Introduction, where we gave a brief overview of some domains of application, and noted that informing was often, though not always, the goal in NLG.

Over the past decade or so, however, there has been a growing trend in the NLG literature to also focus on aspects of textual information delivery that are arguably non-propositional, that is, features of text that are not strictly speaking grounded in the input data, but are related to the manner of delivery. In this section, we focus on these trends, starting with the broad concept of ‘stylistic variation’, before turning to generation of affective text and politeness.

5.1 Generating with style: textual variation and personality

What does the term ‘linguistic style’ refer to? Most work on what we shall refer to as ‘stylistic NLG’ shies away from a rigorous definition, preferring to operationalise the notion in the terms most relevant to the problem at hand.

‘Style’ is usually understood to refer to features of lexis, grammar and semantics that collectively contribute to the identifiability of an instance of language use as pertaining to a specific author, or to a specific situation (thus, one distinguishes between levels of stylistic formality, or speaks of the distinctive characteristics of the style of William Faulkner). This implies that any investigation of style must concern itself, at least in part, with *variation* among the features that mark such authorial or situational variables. In line with this usage, this section reviews developments in NLG in which variation is the key concern, usually at the *tactical*, rather than the strategic, level, the idea being that a given piece of information can be imparted in linguistically distinct, ways (cf. van der Sluis & Mellish, 2010). This strategy was, for example, explicitly adopted by Power et al. (2003).

Given its emphasis on linguistic features, controlling style (however it is defined) is a problem of great interest for NLG since it directly addresses issues of *choice*, which are arguably the hallmark of any NLG system (c.f. Reiter, 2010). Early contributions in this area defined stylistic features using rules to vary generation according to pragmatic goals. For example, McDonald and Pustejovsky (1985) argued that ‘prose style Is a consequence of what decisions are made during the transition from the conceptual representation level to the linguistic level’ (p. 61), thereby placing the problem within the domain of sentence planning and realisation. This stance was also adopted by DiMarco and Hirst (1993), who focus on syntactic variation, proposing a stylistic grammar for English and French.

More recently, a similar perspective was adopted by Walker et al. (2002), in their description of how the SPOT sentence planner was adapted to learn strategies for different communicative goals, as reflected in the rhetorical and syntactic structures of the sentence plans. The planner was trained using a boosting technique to learn correlations between features of sentence plans and human ratings of the adequacy of a sample of outputs for different communicative goals.

Like Walker et al. (2002), contemporary approaches to stylistic variation have tended to eschew rules in favour of data-driven methods to identify relevant features and dimensions of variation from corpora, in what might be thought of as an *inductive* view of style, where variation is characterised by the distribution of whatever linguistic features are considered relevant. An important precedent for this view is Biber’s corpus-based multidimensional approach to style and register variation (Biber, 1988), roughly a contemporary of the grammar-inspired approach of DiMarco and Hirst (1993).

Biber’s model was at the heart of work by Paiva and Evans (2005), which exhibits some characteristics in common with the ‘global’ statistical approaches to NLG discussed in Section 3.3, insofar as it exploits statistics to inform decision-

making at the relevant choice points, rather than to filter the outputs of an overgeneration module. Paiva and Evans (2005) used a corpus of patient information leaflets, conducting factor analysis on their linguistic features to identify two stylistic dimensions. They then allowed their system to generate a large number of texts, varying its decisions at a number of choice points (e.g. choosing a pronoun versus a full NP) and maintaining a trace. Texts were then scored on the two stylistic dimensions, and a linear regression model was developed to predict the score on a dimension based on the choices made by the system. This model was used during testing to predict the best choice at each choice point, given a desired style. Style, however, is a global feature of a text, though it supervenes on local decisions. These authors solved the problem by using a best-first search algorithm to identify the *series* of local decisions as scored by the linear models, that was most likely to maximise the desired stylistic effect, yielding variations such as the following (examples from Paiva & Evans, 2005, p. 61):

- (18) The dose of the patient's medicine is taken twice a day. It is two grams.
- (19) The two-gram dose of the patient's medicine is taken twice a day.
- (20) The patient takes the two-gram dose of the patient's medicine twice a day.

Some authors (e.g., Mairesse & Walker, 2011, , on which more below) have noted that certain features, once selected, may ‘cancel’ or obscure the stylistic effect of other features. This raises the question whether style can in fact be modelled as a linear, additive phenomenon, in which each feature contributes to an overall perception of style independently of others (modulo its weight in the regression equation).

A second question is whether stylistic variation could be modelled in a more specific fashion, for example, by tailoring style to a specific author, rather than to generic dimensions related to ‘formality’, ‘involvement’ and so on. For instance a corpus-based analysis of human-written weather forecasts by Reiter et al. (2005) found that lexical choice varies in part based on the author. One line of work has investigated this using corpora of referring expressions, such as the TUNA Corpus (van Deemter et al., 2012a), in which multiple referring expressions by different authors are available for a given input domain. For instance, Bohnet (2008) and Di Fabrizio et al. (2008) explore statistical methods to learn individual preferences for particular attributes, a strategy also used by Viethen and Dale (2010). Hervás et al. (2013) use case-based reasoning to inform lexical choice when realising a set of semantic attributes for a referring expression, where the case base differentiates between authors in the corpus to take individual lexicalisation preferences into account (see also Hervás et al., 2016).

A more ambitious view of individual variation is present in the work of Mairesse and Walker (2010, 2011), in the context of NLG for dialogue systems. Here, the aim is to vary the output of a generator so as to project different

personality traits. Similar to the model of Biber (1988), personality is here given a multidimensional definition, via the classic ‘Big 5’ model (e.g., John & Srivastava, 1999), where personality is a combination of five major traits (e.g. introversion/extraversion). However, while stylistic variation is usually defined as a linguistic phenomenon, the linguistic features of personality are only indirectly reflected in speaking or writing (a hypothesis underlying much work on detection of personality and other features in text, including Oberlander & Nowson, 2006; Argamon et al., 2007; Schwartz et al., 2013; Youyou et al., 2015).

Mairesse and Walker’s PERSONAGE system, which was developed in the restaurant domain, takes as input a pragmatic goal and, like the system of Paiva and Evans (2005), a list of real-valued style parameters, this time representing scores on the five personality traits. The system estimates generation parameters for stylistic features based on the input traits, using machine-learned models acquired from a dataset pairing sample utterances with human personality judgements. For example, an utterance reflecting high extraversion might be more verbose and involve more use of expletives (see example 21), compared to a more introverted style, which might demonstrate more uncertainty, for example through the use of stammering and hedging (example 22).

- (21) Kin Khao and Tossed are bloody outstanding. Kin Khao just has rude staff. Tossed features sort of unmannered waiters, even if the food is somewhat quite adequate.
- (22) Err... I am not really sure. Tossed offers kind of decent food. Mmhm... However, Kin Khao, which has quite ad-ad-adequate food, is a thai place. You would probably enjoy these restaurants.

An interesting outcome of the evaluation with human subjects reported by Mairesse and Walker (2011) is that readers vary significantly in their judgements of what personality is actually reflected by a given text. This suggests that the relationship between such psychological features and their linguistic effects is far from straightforward. The same could probably be said of stylistic more generally. Clearly, this is an area that is ripe for further research.

5.2 Generating with feeling: affect and politeness

Personality is usually thought of in terms of *traits*, which are relatively stable across time. However, language use may vary not only across individuals, as a function of their stable characteristics, but also within individuals across time, as a function of their more transient affective *states*. ‘Affective NLG’ (a term due to De Rosis & Grasso, 2000) is concerned with variation that reflects emotional states which, unlike personality traits, are relatively transient. In this case, the goals can be twofold: i) to induce an emotional state in the receiver; or (ii) to reflect the emotional state of the producer.

As in the case of personality, the relationship between emotion and language is far from clear, as noted by Belz (2003). For one thing, it isn’t clear whether

only surface linguistic choices need be affected. Some authors have argued that a text's affective impact impinges on content selection; this stance has been adopted, for example, in some applications in e-health where reporting of health-related issues should be sensitive to their potential emotional impact (DiMarco et al., 2007; Mahamood & Reiter, 2011).

Most work on affective NLG has however focussed on tactical choices (e.g. Hovy, 1988; Fleischman & Hovy, 2002; Strong et al., 2007; van Deemter et al., 2008). Various linguistic features that can have emotional impact have been identified, from the increased use of redundancy to enhance understanding of emotionally laden messages (Walker, 1992; De Rosis & Grasso, 2000), to the increased use of first-person pronouns and adverbs, as well as sentence ordering to achieve emphasis or reduce adverse emotional impact (De Rosis & Grasso, 2000).

This research on affective NLG relies on models of emotion of various degrees of complexity and cognitive plausibility. The common trend underlying all these approaches however is that emotional states should impact lexical, syntactic and other linguistic choices. The question then is to what extent such choices are actually perceived by readers or users of a system.

In an empirical study, van der Sluis and Mellish (2010) reported on two experiments investigating the effect of various tactical decisions on the emotional impact of text on readers. In one experiment, texts gave a (fake) report to participants on their performance on an aptitude test, with manually induced variations, such as these:

- (23) *Positive slant*: On top of this you also outperformed most people in your age group with your exceptional scores for Imagination and Creativity (7.9 vs 7.2) and Logical- Mathematical Intelligence (7.1 vs. 6.5).
- (24) *Neutral/factual slant*: You did better than most people in your age group with your scores for Imagination and Creativity (7.9 vs 7.2) and Logical-Mathematical Intelligence (7.1 vs. 6.5).

Evaluation of these texts showed that the extent to which affective tactical decisions influence hearer's emotional states is dependent on a host of other factors, including the degree to which the reader is directly implicated in what the text says (in the case of an aptitude test, the reader would be assumed to feel the outcomes have personal relevance). An important question raised by this study is how affect should be measured: van der Sluis and Mellish (2010) used a standardised self-rating questionnaire to estimate changes in affect before and after reading a text, but the best way to measure emotion remains an open question.

The emotional slant in the language used by an author or speaker may have implications for the degree to which the listener or reader may feel 'impinged upon'. This becomes particularly relevant in interactive systems, where NLG components are generating language in the context of dialogue. Consider, for example, the difference between these requests:

- (25) *Direct strategy*: Chop the tomatoes!
- (26) *Approval strategy*: Would it be possible for you to chop the tomatoes?
- (27) *Autonomy strategy*: Could you possibly chop the tomatoes?
- (28) *Indirect strategy*: The tomatoes aren't chopped yet.

The four strategies exemplified above come across as having varying degrees of politeness which, according to one influential account (Brown & Levinson, 1987), depends on *face*. *Positive face* reflects the speaker's desire that some of her goals be shared with her interlocutors; *negative face* refers to the speaker's desire not to have her goals impinged upon by others. The connection with affect that we suggested above hinges on these distinctions: different degrees of politeness reflect different degrees of 'threat' to the listener; hence, generating language based on the right face strategy could be seen as a branch of affective NLG.

In an early, influential proposal, Walker et al. (1997) proposed an interpretation of (Brown & Levinson, 1987) in terms of the four dialogue strategies, exemplified in (25 – 28) above. Subsequently, Moore et al. (2004) used this framework in the generation of tutorial feedback, where a discourse planner used a Bayesian network to inform linguistic choices compatible with the target politeness/affect value in a given context (see Johnson et al., 2004, for a related approach).

Gupta et al. (2007) also used the four dialogue strategies identified by Walker et al. (1997) in the POLLY system, which used STRIPS-based planning to generate a plan distributed among two agents in a collaborative task (see also Gupta et al., 2008). An interesting finding in their evaluation is that perception of face-threat depends on the speech act; for example, requests can be more threatening. Gupta et al. (2007) also note possible cultural differences in perception of face threat (in this case, between UK and Indian participants).

5.3 Style and affect: concluding remarks

In summary, work on variation in NLG, mostly focussing on tactical decisions, or surface-level linguistic features, has sought to model differences of register or style as well as individual differences due to personality and affective state.

This area of NLG research is still in a rather fledgling state, with several open questions of both theoretical and computational import. Among these is the question of how best to model complex, multi-dimensional constructs such as personality or emotion; this question speaks both to the cognitive plausibility of the models informing linguistic choices, and to the practical viability of different machine learning strategies that could be leveraged for the task (for example, linear, additive models versus more 'global' models of personality or style). Also important here is the kind of data used to inform generation strategies: as we have seen above, a lot of affective NLG work relies on ratings by human judges. However, some recent work in affective computing has questioned the

use of ratings, comparing them to ranking-based and physiological methods (e.g. Martinez et al., 2014; Yannakakis & Martínez, 2015). This and similar research is probably of high relevance to NLG researchers.

A second important question is which linguistic choices truly convey the intended variation to the reader or listener. While current systems use a range of devices, from aggregation strategies to lexical choice, it is not clear which ones are actually perceived as having the desired effect.

A third important research avenue, which is especially relevant to interactive systems, is adaptivity, that is, the way speakers (or systems) alter their linguistic choices as a result of their interlocutors' utterances (Clark, 1996; Niederhoffer & Pennebaker, 2002; Pickering & Garrod, 2004), a theme that has also begun to be explored in NLG

6 Generating creative and entertaining text

‘Good’ writers not only present their ideas in coherent and well-structured prose. They also succeed in keeping the attention of the reader through narrative techniques, and in occasionally surprising the reader, for example, by creative language use such as small jokes or well-placed metaphors (see e.g., among many others, Flower & Hayes, 1981; Nauman et al., 2011; Veale & Li, 2015). The NLG techniques and applications that we discussed so far in this survey, arguably, do not simulate good writers in this sense, and as a result automatically generated texts can be perceived as somewhat boring and repetitive.

This lack of attention to creative aspects of language production within NLG is not due to a general lack of scholarly interest in these phenomena. Indeed, computational research into creativity has a long tradition, with roots that go back to the early days of AI (as Gervás, 2013, notes, the first story generation algorithm on record, Novel Writer, was developed by Sheldon Klein in 1973). However, it is fair to say that, so far, there has been little interaction between researchers from the computational creativity and NLG communities respectively, even though both groups in our opinion could learn a lot from each other. In particular, NLG researchers stand to benefit from insights into what constitutes creative language production, as well as structural features of narrative that have the potential to improve NLG output even in data-to-text systems (see Reiter et al., 2008, for an argument to this effect in relation to a medical text generation system). At the same time, researchers in computational creativity could also benefit from the insights provided by the NLG community where the generation of fluent language is concerned since, as we shall see, a lot of the focus in this research, especially where narrative is concerned, is on the generation of plans and on content determination.

In what follows, we give an overview of automatic approaches to creative language production, starting from relatively simple jokes and metaphors to more advanced forms, such as narratives.

6.1 Generating puns and jokes

Consider:

- (29) What's the difference between money and a bottom?
One you spare and bank, the other you bare and spank.
- (30) What do you call a weird market?
A bizarre bazaar.

These two (pretty good!) *punning riddles* were automatically generated by the JAPE system developed by Binsted and Ritchie (1994, 1997). Punning riddles form a specific joke genre and have received considerable attention in the context of computational humor, presumably because they are relatively straightforward to define, often relying on spelling or word sense ambiguities. Many good, human-produced examples have been collected in joke books and sites and may thus act as a source of inspiration.

Simplifying somewhat, JAPE (Joke Analysis and Production Engine) relies on a template-based NLG system, combining fixed text (*What's the difference between X and Y?* or *What do you call X?*) with slots, which are the source of the riddle. Various standard lexical resources are used for joke production, including a British pronunciation dictionary (to find different words with a similar pronunciation, such as ‘bizarre’ and ‘bazaar’) and WordNet (Miller, 1995, to find words with a similar meaning, such as *bazaar* and *market*). JAPE uses various techniques to create the punning riddles, such as juxtaposition, in which related words are simply placed next to each other and treated as a normal construction, while making sure that the combination is novel (i.e., not in the JAPE database already). It is interesting to observe that in this way JAPE may automatically come up with existing jokes (a quick Google search reveals that many bizarre bazaars, as well as bazaar bizarres, exist).

NLG evaluation is difficult, in general (as we will discuss in more detail in Section 7 below) and evaluation of humorous NLG is arguably even harder. Nevertheless, Binsted et al. (1997) showed that it can be done, in an elegant manner. They presented 120 8-11 year old children with a number of punning riddles, some automatically generated by JAPE and some selected from joke books. They also included a number of non-joke controls, such as:

- (31) What do you get when you cross a horse and a donkey?
A mule

For each stimulus that they were exposed to, children were asked to indicate whether they thought it was a joke, and how funny they considered it. The results revealed that computer generated riddles were recognised as jokes, and considered funnier than non-jokes. Interestingly, the joke children rated highest was automatically generated by JAPE (we urge the reader to inspect the original paper), although in general, human-produced jokes were considered funnier by children than automatically generated ones.

Following the seminal work of Binsted and Ritchie, various other systems have been developed which can automatically generate jokes, including for example the HAHAcronym system of Stock and Strapparava (2005), which produces humorous acronyms, and the system of Binsted et al. (2003), which focusses on the generation of referential jokes (“It was so cold, I saw a lawyer with his hands in his own pockets.”).

Petrovic and Matthews (2013) offer an interesting, unsupervised alternative to this earlier work, which does not require labelled examples or hard-coded rules. Like their predecessors, Petrovic and Matthews also start from a template – in their case *I like my X like I like my Y, Z* – where *X* and *Y* are nouns (e.g., *coffee* and *war*) and *Z* is an attribute (e.g., *cold*). Clearly, linguistic realisation is not an issue, but content selection – finding ‘funny’ triples *X, Y* and *Z* – is a challenge. Interestingly, the authors postulate a number of guiding principles for ‘good’ triples. In particular, they hypothesize that (a) the joke is funnier if the attribute *Z* can be used to describe both nouns *X* and *Y*; (b) the joke is funnier if attribute *Z* is both common and ambiguous; and (c) the joke is funnier the more dissimilar *X* and *Y* are. These three statements can be quantified relying on standard resources such as Wordnet and the Google n-gram corpus (Brants & Franz, 2006), and using these measures their system outputs, for example:

- (32) I like my relationships like I like my source, open.

Again, evaluation is tricky – but interesting. The authors harvested human-written jokes from Twitter, conforming to the same *I like my X ...* template, after which their different models are used to generate new jokes, not found anywhere on line, with the same *X* but different *Y* and *Z*. Human judges then blindly rated both the human and the model-generated jokes. Results showed that the jokes of the best model were rated as funny in 16% of the cases. This may not seem like much, but it should be taken into account that whether you think something is funny or not is presumably very personal; indeed only 33% of the human jokes were considered funny.

It is probably fair to say that computational joke generation research to date has mostly focussed on laying bare the basic structure of certain relatively simple puns and exploiting these to good effect (e.g., Ritchie, 2009). However, many other kinds of jokes exist, often requiring sophisticated, hypothetical reasoning. Presumably, many of the central problems within AI need to be solved first before generation systems will be capable of producing these kinds of advanced jokes.

6.2 Generating metaphors and similes

Whether you think something is funny or not may be subjective, but in any case insights from joke generation can be useful as a stepping stone towards a better understanding of creative language use, including metaphor, simile and analogy. In all of these, a mapping is made between two conceptual domains, in such a way that terminology from the source domain is used to say something

about the target domain, typically in a nonliteral fashion, which can be helpful in computer-generated texts to illustrate complex information. For example, Hervás et al. (2006) study analogies in narrative contexts, such as *Luke Skywalker was the King Arthur of the Jedi Knights*, which immediately clarifies an important aspect of Luke Skywalker for those not in the know. In a simile, the two domains are compared (A ‘is like’ B); in a metaphor they are equated. Jokes and metaphors/similes are related: the automatically generated jokes of Petrovic and Matthews are comparable to similes, while Kiddon and Brun (2011), for example, frame the problem of identifying double entendre jokes as a type of metaphor identification. Nevertheless, one could argue that generating jokes is more complex because of the extra funniness constraint.

Like computational humor, the automatic recognition and interpretation of metaphorical, non-literal language has received considerable attention since the early days of AI (see Shutova et al., 2012, for an overview). Martin (1990, 1994), for example, focussed on the recognition of metaphor in the context of Unix Support, as in the following examples:

- (33) How can I kill a process?
- (34) How can I enter lisp?

The first one, for example, makes a mapping between ‘life’ (source) and ‘processes’ (target), and is by now so common that is almost a dead metaphor, but this was not the case in the early days of Unix. Clearly, understanding of the metaphors is a prerequisite for automatically answering these questions. Early research on the computational interpretation of metaphor already recognised that metaphors rely on semantic conventions that are exploited (‘broken’) to express new meanings. A system for metaphor understanding, as well as one for metaphor generation, therefore requires knowledge about what literal meanings are, and how these can be stretched or translated into metaphoric meanings (e.g., Wilks, 1978; Fass, 1991).

Recent work by Veale and Hao (Veale & Hao, 2007, 2008) has shown that this kind of knowledge can be acquired from the web, and used for the generation of new metaphors and similes (comparisons). Their system, called Sardonicus, is capable of generating metaphors for user-provided targets (T), such as the following, expressing that Paris Hilton (‘the person, not the hotel, though the distinction is lost on Sardonicus’ Veale & Hao, 2007, p.1474) is skinny:

- (35) Paris Hilton is a stick

Sardonicus searches the web for nouns (N) that are associated with skinniness, which are included in a case-base and range from *pole*, *pencil*, and *stick* to *snake* and *stick insect*. Inappropriate ones (like *cadaver*) are ruled out, based on the theory of category-inclusion of Glucksberg (2001). This list of potential similes is then used to create Google queries, inspired by the work of Hearst (1992), of the form N-like T (e.g., *stick insect-like Paris Hilton*, which actually occurs on the web), giving a ranking of the potential similes to be generated.

A comparable technique is used by Veale (2013) to generate metaphors with an affective component, as in ‘Steve Jobs was a great leader, but he could be such a tyrant’. The Google n -gram corpus is used to find stereotypes suitable for simile generation (e.g., ‘lonesome as a cowboy’), a strategy reminiscent of the use of web-scale n -gram data to smooth the output of image-to-text systems (see Section 4). Next, an affective dimension is added, based on the assumption that properties occurring in a conjunction (‘as *lush and green* as a jungle’) are more likely to have the same affect than properties that do not. Using positive (e.g., ‘happy’, ‘wonderful’) and negative (e.g., ‘sad’, ‘evil’) seeds, coordination queries (e.g., ‘happy and X’) are used to collect positive and negative labels for stereotypes, indicating, for instance, that babies are positively associated with qualities such as ‘smiling’ and ‘cute’, and negatively associated with ‘crying’ and ‘sniveling’. This enables the automatic generation of positive (‘cute as a baby’) and negative (‘crying like a baby’) similes. Veale (2013) even points out that by collecting, for example, a number of negative metaphors for Microsoft being a monopoly, and using these in a set of predefined tropes, it becomes possible to automatically generate a poem such as the following:

No Monopoly Is More Ruthless

Intimidate me with your imposing hegemony
No crime family is more badly organized, or controls more ruthlessly
Haunt me with your centralized organization
Let your privileged security support me
O Microsoft, you oppress me with your corrupt reign

In fact, automatic generation of poetry is an emerging area at the crossroads of computational creativity and natural language generation (see for example Lutz, 1959; Gervás, 2001; Wong et al., 2008; Netzer et al., 2009; Greene et al., 2010; Colton et al., 2012; Manurung et al., 2012; Zhang & Lapata, 2014b, for variations on this theme).

6.3 Generating narratives

Computational narratology is concerned with computational models for the generation and interpretation of narrative texts (e.g., Gervás, 2009; Mani, 2010, 2013). The starting point for many approaches to narrative generation is a view of narrative coming from classical narratology, a branch of literary studies with roots in the Formalist and Structuralist traditions (e.g., Propp, 1968; Genette, 1980; Bal, 2009). This field has been concerned with analysing both the defining characteristics of narrative, such as plot or character, and more subtle features, such as the handling of time and temporal shifts, focalisation (that is, the ability to convey to the reader that a story is being recounted from a specific point of view), and the interaction of multiple narrative threads, in the form of sub-plots, parallel narratives, etc. An important recent development is the interest, on the part of narratologists, in bringing to bear insights from Cognitive Science and AI on their literary work, making this field ripe for multi-disciplinary interaction

(see especially Herman, 1997, 2007; Meister, 2003, for programmatic statements to this effect, as well as theoretical contributions).

Classical narratology makes a fundamental distinction between the ‘story world’ and the text that narrates the story. In line with the formalist and structuralist roots of this tradition, the distinction is usually articulated as a dichotomy between *fabula* (or story) and *suzjet* (or discourse). There is a parallel between this distinction and that between a text plan in NLG, versus the actual text which articulates that plan. However, the crucial difference is that in producing a plan for a narrative, a story generation system typically does not use input data of the sort required by most of the NLG systems reviewed thus far, since the story is usually fictional. On the other hand, narratological tools have also been successfully applied to real-world narratives, including oral narratives of personal experience (e.g., Herman, 2001; Labov, 2010).

The focus of most work on narrative generation has been on the pre-linguistic stage, that is, on generating plans within a story world for fictional narratives, usually within a specific genre whose structural properties are well-understood, for example, fairy tales or Arthurian legends (see Gervás, 2013, for a review). There are however links between the techniques used for such stories and those we have discussed above in relation to NLG (see especially Section 3.2). Prominent among these are planning and reasoning techniques to model the creative process as a problem-solving task. For example, MINSTREL (Turner, 1992) uses reasoning to model creativity from the author’s perspective, producing narrative plans based on authorial goals, such as the goal of introducing drama into a narrative, while ensuring thematic consistency.

More recently, BRUTUS (Bringsjord & Ferrucci, 1999) used a knowledge base of story schemas, from which one is selected and elaborated using planning techniques to link causes and effects (see also Young, 2008; Riedl & Young, 2010, among others, for recent examples of the use of planning techniques to model the creative process in narrative generation).

As Gervás (2010) notes, the focus on planning story worlds and modelling creativity has often implied a sidelining of linguistic issues, so that rendering a story plan into text has often been viewed as a secondary consideration. For example Figure 7a shows an excerpt of a story produced by the TALESPLAN system (Meehan, 1977): here, the emphasis is on using problem-solving techniques to produce a narrative in which events follow from each other in a coherent fashion, rather than on telling it in a fluent way. An important exception to this trend is the work of Callaway and Lester (2002), who explicitly addressed the gap between computational narratology and NLG. Their system took a narrative plan as a starting point, but focussed on the process of rendering the narrative in fluent English, handling time shifts, aggregation, anaphoric NPs and many other linguistic phenomena, as the excerpt in Figure 7b shows.⁹ In addition, there have been a number of contributions from the generation community on more specific issues related to narrative, such as how to convey the temporal

⁹It is worth noting that this system has since been re-used in the context of generating interactive text for a portable museum guide by Stock et al. (2007).

John Bear is somewhat hungry. John Bear wants to get some berries. John Bear wants to get near the blueberries. John Bear walks from a cave entrance to the bush by going through a pass through a valley through a meadow. John Bear takes the blueberries. John Bear eats the blueberries. The blueberries are gone. John Bear is not very hungry.

Once upon a time a woodman and his wife lived in a pretty cottage on the borders of a great forest. They had one little daughter, a sweet child, who was a favorite with every one. She was the joy of her mother's heart. To please her, the good woman made her a little scarlet cloak and hood. She looked so pretty in it that everybody called her Little Red Riding Hood.

(a) Excerpt from TALESPIN (Meehan, 1977) (b) Excerpt from STORYBOOK (Callaway & Lester, 2002)

Figure 7: Examples of automatically generated narratives. The left panel shows an excerpt from a story produced by TALESPIN (Meehan, 1977); the right panel is an excerpt from the *Little Red Riding Hood* fairy-tale, generated by the STORYBOOK system (Callaway & Lester, 2002).

flow of narrative discourse (Oberlander & Lascarides, 1992; Dorr & Gaasterland, 1995; Elson & McKeown, 2010). This is a problem that deserves more attention in NLG, since texts with a complex narrative structure often narrate events in a different order from which they occurred. For example, events may be planned in order of importance, rather than temporally, even when they are grounded in real-world data (e.g. Portet et al., 2009). This makes the use of the right choices for tense, aspect and temporal adverbials crucial to ensure clarity for the reader. This type of complexity in narrative structure also emerges in interactive narrative fiction (for example, in games; cf., Montfort, 2007).

Beyond the focus on specific linguistic issues, there has also been some work that leverages data-driven techniques to generate stories. For example, McIntrye and Lapata (2009) propose a story generation system whose input is a database of entities and their interactions, extracted from a corpus of stories by parsing them, retrieving grammatical dependencies, and building chains of events in which specific entities play a role. The outcome is a graph encoding a partial order of events, with edges weighted by mutual information to reflect the degree of association between nodes. Sentence planning then takes place using template-like grammar rules specifying verbs with subcategorisation information, followed by realisation using REALPRO (Lavoie & Rambow, 1997). One of the most interesting features of this work is the coupling of the generation model with an interest model to predict which stories would actually be rated as interesting by readers. This was achieved by training a kernel-based classifier on

shallow lexical and syntactic features of stories, a novel take on an old problem in narratology, namely, what makes a story ‘tellable’, thereby distinguishing it from a mere report (e.g., Herman, 1997; Norrick, 2005; Bruner, 2011).

Most story generation work is restricted to (very) short stories. It is certainly true that planning a book-length narrative along the lines sketched above is extremely challenging, but researchers have recently started exploring the possibilities, for instance in the context of NaNoGenMon (National Novel Generation Month), in which participants write a computer program capable of generating a ‘novel’. Perhaps the best known example is *World Clock* (Montfort, 2013) which describes 1440 (24 × 60) events taking place around the world, one randomly selected minute at a time. These are the first two:

It is now exactly 05:00 in Samarkand. In some ramshackle dwelling a person who is called Gang, who is on the small side, reads an entirely made-up word on a box of breakfast cereal. He turns entirely around.

It is now right about 18:01 in Matamoros. In some dim yet decent structure a man named Tao, who is no larger or smaller than one would expect, reads a tiny numeric code from a recipe clipping. He smiles a tiny smile.

The book was fully generated by 165 lines of Python code, written by the author in a few hours, and later published (together with the software) by Harvard Book Store press. There is even a Polish translation (by Piotr Marecki), created by translating the Python algorithm.

Turning to evaluation of narrative generators, this is an area where the consensus in the field is that much further research effort is required (see Zhu, 2012, for a recent argument to this effect). To some extent, the problems faced in evaluating story generators mirror those in classical NLG. For example, evaluating content determination while factoring out the impact of sentence planning and realisation is far from trivial (see Mellish & Dale, 1998, and the discussion of evaluation in Section 7). However, in the case of fictional narrative, the problem is exacerbated by the fact that there is usually no ‘objective’ input against which to compare a story plan; indeed, the main focus of evaluation here is the success with which a system models the creative process, constructing stories that have such qualities as novelty (e.g., Pérez et al., 2011) or believability of characters (e.g., Riedl & Young, 2005). Where the focus is on narrative *language*, evaluation is more clearly oriented towards linguistic issues such as coherence and fluency, as shown by the exhaustive evaluation conducted by Callaway and Lester (2002) for the STORYBOOK system.

The problem for those systems in which linguistic quality is secondary is that it is difficult to evaluate a story while factoring out effects which are due to the way it is recounted. Recent proposals, for example by Rowe et al. (2009), have emphasised the need to deploy multiple evaluation methods to evaluate narratives at different levels, from plot structure to cognitive-affective impact on readers. As we shall see in Section 7, the use of multiple methods is probably desirable even for classical NLG tasks.

6.4 Generating creative language: Concluding remarks

In this section we have highlighted recent developments in the broad area of creative language generation, a topic which is rather understudied in NLG. Nevertheless, we would like to argue that NLG researchers can improve the quality of their output by taking insights from computational creativity on board.

Work that exploits corpora and other lexical resources for the automatic generation of jokes, puns, metaphors and similes has revealed different ways in which words are related and can be juxtaposed to form unexpected and possibly even ‘funny’ or ‘poetic’ combinations. Given that, for example, metaphor is pervasive in everyday language (as argued, for example, by Lakoff & Johnson, 1980), not just in overtly creative uses, NLG researchers interested in enhancing the readability – and especially the variability – of the text-generating capability of their models would benefit from a closer look at work in poetry, joke and metaphor generation.

In a similar vein, work on narratology is rich in insights on the interaction of multiple threads in a single narrative, and how the choice of events and their ordering can give rise to interesting stories (e.g., Gervás, 2012). These insights are valuable, for example, in the development of more elaborate text planners in domains where time and causality play a role. Similarly, narratological work on character and focalisation can also help in the development of better NLG techniques to vary output according to specific points of view, an area that we touched on in Section 5,

We have seen that evaluation of these systems remains something of a bottleneck. In part, this is because it is not always easy to determine the ‘right’ question to ask in an evaluation. For instance, as we saw in the case of joke and poetry generators, demonstrating genre compatibility and recognition (‘Is this a joke?’) is arguably already an achievement, insofar as it suggests that a system is producing artefacts that conform to normative expectations. At the same time, the emphasis on creativity, especially in story generation systems, suggests that conformity to genre conventions is not the only question at stake. The problem is that evaluation is difficult to carry out without ensuring quality at *all* levels of the generation process, from planning to realisation. This is probably an area in which NLG has much to offer to computational creativity researchers, in the form of a set of techniques to ensure coherence and fluency.

7 Evaluation

Though we have touched on the subject of evaluation at various points, it deserves a full discussion as a topic which has become a central methodological concern in NLG. A factor that contributed to this development was the establishment of a number of NLG shared tasks, launched in the wake of an NSF-funded workshop held in Virginia in 2007 (Dale & White, 2007). These tasks have focussed on referring expression generation (Belz et al., 2010; Gatt & Belz, 2010); surface realisation (Belz et al., 2011); generation of instructions in virtual

environments (Striegnitz et al., 2011; Janarthanam & Lemon, 2011); content determination (Bouayad-Agha et al., 2013; Banik et al., 2013); and question generation (Rus et al., 2011). Recent proposals for new challenges extend these to narrative generation (Concepción et al., 2016), generation from structured web data (Colin et al., 2016) and from unaligned pairs of meaning representations and text (Novikova & Rieser, 2016a). In image captioning, shared tasks have helped the development of large-scale datasets and evaluation servers such as MS-COCO¹⁰ (cf. Section 4.1). In general, however, NLG evaluation is marked by a great deal of variety and it is difficult to compare systems directly. There are at least two reasons why this is the case.

Variable input There is no single, agreed-upon input format for NLG systems (McDonald, 1993; Mellish & Dale, 1998; Evans et al., 2002). Typically, one can only compare systems against a common benchmark if the input is similar. Examples are the image-captioning systems described in Section 4, or systems submitted to one of the shared tasks mentioned above. Even in case a common ‘standard’ dataset is available for evaluation, comparison may not be straightforward due to input variation, or due to implicit biases in the input data. For example, Rajkumar and White (2014) observe that, despite many realisers being evaluated against the Penn Treebank, they make different assumptions about the input format, including how detailed the pre-syntactic input representation is, a problem also observed in the first Surface Realisation shared task (Belz et al., 2011). As Rajkumar and White (2014) note, a comparison of realisers on the basis of scores on the Penn Treebank shows that the highest-ranking is the FUF/SURGE realiser (which is second in terms of coverage), based on experiments by Callaway (2005). However, these experiments required painstaking effort to extract the input representations at the level of detail needed by FUF/SURGE; other realisers support more underspecified input. In a related vein, image captioning evaluation studies have shown that many datasets contain a higher proportion of nouns than verbs, and few abstract concepts (Ferraro et al., 2015), making systems that generate descriptions emphasising objects more likely to score better. The relevance of this observation is shown by Elliott and De Vries (2015), who note that the ranking of their image captioning system based on visual dependency grammar depends in part on the data it is evaluated on, with better performance on data containing more images depicting actions (we return to this study below).

Multiple possible outputs Even for a single piece of input and a single system, the range of possible outputs is open-ended, a problem that arguably holds for any NLP task involving textual output, including machine translation and summarisation. Corpora often display a substantial range of variation and it is often unclear, without an independent assessment, which outputs are to be preferred (Reiter & Sripada, 2002). In the image captioning literature, authors who have framed the problem in terms of retrieval have motivated the choice in part

¹⁰<http://mscoco.org/dataset/#captions-upload>

based on this problem, arguing that ‘since there is no consensus on what constitutes a good image description, independently obtained human assessments of different caption generation systems should not be compared directly’ (Hodosh et al., 2013, p. 580). While capturing variation may itself be a goal (e.g., Belz, 2008; Viethen & Dale, 2010; Hervás et al., 2013; Ferreira et al., 2016), as we also saw in our discussion of style in Section 5, this is not always the case. Thus, in a user-oriented evaluation, the SUMTIME-MOUSAM system weather forecasts were preferred by readers over those written by forecasters because the latter’s lexicalisation decisions were susceptible to apparently arbitrary variation (Reiter et al., 2005); similar outcomes were more recently reported for statistical NLG systems trained on the SUMTIME corpus (Belz, 2008; Angeli et al., 2010).

Rather than give an exhaustive review of NLG evaluation – hardly a realistic prospect given the diversity we have pointed out – the rest of this section will highlight some topical issues in current work. By way of an overview of these issues, consider the hypothetical scenario sketched in Figure 8, which is loosely inspired by work on various weather-reporting systems developed in the field. This NLG system is embedded in the environment of an offshore oil-rig; the relevant features of the *setup* (in the sense of Sparck Jones & Galliers, 1996) are the system itself and its users, here a group of engineers. While the *task* of the system is to generate weather reports from numerical weather prediction data, its ultimate *purpose* is to facilitate users’ planning of drilling and maintenance operations. Figure 8 highlights some of the common questions addressed in NLG evaluation, together with a broad typology of the methods used to address them, in particular, whether they are objective – that is measurable against an external criterion, such as corpus similarity or experimentally obtained behavioural data – or subjective, requiring human judgements.

A fundamental methodological distinction, due to Sparck Jones and Galliers (1996), is between *intrinsic* and *extrinsic* evaluation methods. In the case of NLG, an intrinsic evaluation measures the performance of a system without reference to other aspects of the setup, such as the system’s effectiveness in relation to its users. In our example scenario, questions related to text quality, correctness of output and readability qualify as intrinsic, whereas the question of whether the system actually achieves its goal in supporting adequate decision-making on the offshore platform is extrinsic.

7.1 Intrinsic methods

Intrinsic evaluation in NLG is dominated by two methodologies, one relying on human judgements (and hence subjective), the other on corpora.

7.1.1 Subjective (human) judgements

Human judgements are typically elicited by exposing naive or expert subjects to system outputs and getting them to rate them on some criteria. Common criteria include:

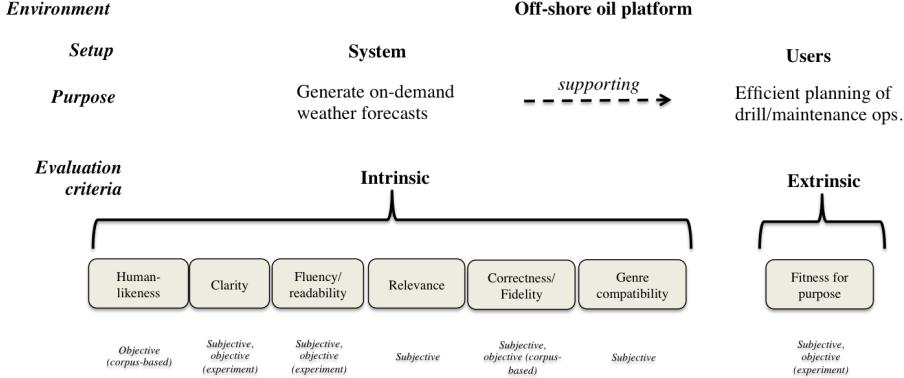


Figure 8: Hypothetical evaluation scenario: a weather report generation system embedded in an offshore oil platform environment. Possible evaluation methods, focussing on different questions, are highlighted at the bottom, together with the typical methodological orientation (subjective/objective) adopted to address them.

- *Fluency or readability*, that is, the linguistic quality of the text (e.g., Callaway & Lester, 2002; Mitchell et al., 2012; Stent et al., 2005; Lapata, 2006; Cahill, 2009; Espinosa et al., 2010, *inter alia*);
- *Accuracy, adequacy, relevance* or *correctness* relative to the input, reflecting the system's rendition of the content (e.g. Lester & Porter, 1997; Srivada et al., 2005; Hunter et al., 2012), a criterion often used in subjective evaluations of image-captioning systems as well (e.g. Kulkarni et al., 2011; Mitchell et al., 2012; Kuznetsova et al., 2012; Elliott & Keller, 2013).

Though they are the most common, these two sets of criteria do not exhaust the possibilities. For example, subjective ratings have also been elicited for argument effectiveness in a system designed to generate persuasive text for prospective house buyers (Carenini & Moore, 2006). In image captioning, at least one system was evaluated by asking users to judge the creativity of the generated caption, with a view to assessing the contribution of web-scale n-gram language models to the captioning quality (Li et al., 2011). Below, we also discuss judgements of genre compatibility (Section 7.1.3).

The use of scales to elicit judgements raises a number of questions. One has to do with the nature of the scale itself. While discrete, ordinal scales are the dominant method, a continuous scale – for example, one involving a visually presented slider (Gatt & Belz, 2010; Belz & Kow, 2011) – might give subjects the possibility of giving more nuanced judgements. For example, a text generated by our hypothetical weather report system might be judged so disfluent as to

be given the lowest rating on an ordinal scale; if the following text is judged as being worse, a subject would have no way of indicating this. A related question is whether subjects find it easier to *compare* items rather than judge each one in its own right. This question has begun to be addressed in the NLP evaluation literature, usually with binary comparisons, for example between the outputs of two MT systems (see Dras, 2015, for discussion). In a recent study evaluating causal connectives produced by an NLG system, Siddharthan and Katsos (2012) used Magnitude Estimation, whereby subjects are not given a predefined scale, but are asked to choose their own and proceed to make comparisons of each item to a ‘modulus’, which serves as a comparison point throughout the experiment (see Bard et al., 1996).¹¹ Belz and Kow (2010) compared a preference-based paradigm to a standard rating scale to evaluate systems from two different domains (weather reporting and REG), and found that the former was more sensitive to differences between systems, and less susceptible to variance between subjects.

An additional concern with subjective evaluations is inter-rater reliability. Multiple judgements by different evaluators may exhibit high variance, a problem that was encountered in the case of Question Generation (Rus et al., 2011). Recently, Godwin and Piwek (2016) suggested that such variance can be reduced by an iterative method whereby training of judges is followed by a period of discussion, leading to the updating of evaluation guidelines. This, however, is more costly in terms of time and resources.

It is probably fair to state that, these days, subjective, human evaluations are often carried out via online platforms such as Amazon Mechanical Turk¹² and CrowdFlower¹³, though this is probably more feasible for widely-spoken languages such as English. A seldom-discussed issue with such platforms concerns their ethical implications (for example, they involve large groups of poorly paid individuals; see Fort et al., 2011) as well as the reliability of the data collected, though measures can be put in place to ensure, for instance, that contributors are fluent in the target language (see e.g., Goodman et al., 2013; Mason & Suri, 2012).

7.1.2 Objective humanlikeness measures using corpora

Intrinsic methods that rely on corpora can generally be said to be addressing the question of ‘humanlikeness’, that is, the extent to which the system’s output matches human output under comparable conditions. From the developer’s perspective, the selling point of such methods is their cheapness, since they are usually based on automatically computed metrics. A variety of corpus-based metrics, often used earlier in related fields such as Machine Translation

¹¹The modulus is an item – a text, or a sentence – which is selected in advance and which subjects are asked to rate first. All subsequent ratings or judgements are performed in comparison to this modus item. Though subjects are able to use any scale they choose, this method allows all judgements to be normalised by the judgement given for the modulus. Typically, normalised judgements are analysed on a logarithmic scale.

¹²<https://www.mturk.com/mturk/welcome>

¹³<https://www.crowdflower.com>

Metric	Description	Origins
N-gram overlap	BLEU	Precision score over variable-length n-grams, with a length penalty (Papineni et al., 2002) and, optionally, smoothing (Lin & Och, 2004).
	NIST	A version of BLEU with higher weighting for less frequent n -grams and a different length penalty (Doddington, 2002).
	ROUGE	Recall-oriented score, with options for comparing non-contiguous n -grams and longest common subsequences (Lin & Hovy, 2003).
	METEOR	Harmonic mean of unigram precision and recall, with options for handling (near-synonymy) and stemming (Lavie & Agarwal, 2007).
	GTM	General Text Matcher. F-Score based on precision and recall, with greater weight for contiguous matching spans (Turian et al., 2003)
	CIDEr	Cosine-based n-gram similarity score, with n-gram weighting using TF-IDF (Vedantam et al., 2015).
String distance	Edit distance	Number of insertions, deletions, substitutions and, possibly, transposition required to transform the candidate into the reference string (Levenshtein, 1966).
	TER	Translation edit rate, a version of edit distance (Snover et al., 2006).
	TERP	Version of TER handling phrasal substitution, stemming and synonymy (Snover et al., 2006).
	TERPA	Version of TER optimised for correlations with adequacy judgments (Snover et al., 2006).
Content overlap	Dice/Jaccard	Set-theoretic measures of overlap between two unordered sets (e.g. of predicates or other content units)
	MASI	Measure of agreement between set-valued items, a weighted version of Jaccard (Passonneau, 2006)
	SPICE	Measure of overlap between candidate and reference texts based on propositional content obtained by parsing the text into graphs representing objects and relations, by first parsing captions into scene graphs representing objects and relations (Anderson et al., 2016)

Table 1: Intrinsic, corpus-based metrics based on string overlap, string distance, or content overlap. The last column indicates the NLP sub-discipline in which a metric originated, where applicable. Legend: MT = Machine translation; AS = automatic summarisation; IC = image captioning.

or Summarisation, have been used in NLG evaluation. Some of the main ones are summarised in Table 1, which groups them according to their principal characteristics, and for each adds a key reference.

Measures of n-gram overlap or string edit distance, usually originating in

Machine Translation or Summarisation (with the exception of CIDEr, Vedantam et al., 2015) are frequently used for evaluating surface realisation (e.g., White et al., 2007; Cahill & Josef, 2006; Espinosa et al., 2010; Belz et al., 2011) and occasionally also to evaluate short texts characteristic of data-driven systems in domains such as weather reporting (e.g. Reiter & Belz, 2009; Konstas & Lapata, 2013) and image captioning (see Bernardi et al., 2016). Edit distance metrics have been exploited for realisation (Espinosa et al., 2010), but also for REG (Gatt & Belz, 2010).

The focus of these metrics is on the text, rather than its fidelity to the input. In a limited number of cases, surface-oriented metrics have been used to evaluate the adequacy with which output text reflects content (Banik et al., 2013; Reiter & Belz, 2009). However, if content determination is the focus, a measure of surface overlap is at best a proxy, relying on an assumption of a straightforward correspondence between input and output. This assumption may be tenable if texts are brief and relatively predictable. In some cases, it has been possible to use metrics to measure content determination directly, based on semantically annotated corpora. For instance, REG algorithms have been evaluated in this fashion using set overlap metrics (Viethen & Dale, 2007; van Deemter et al., 2012a). Direct measurements of content overlap between generated and candidate outputs will likely increase, as automatic data-text alignment techniques make such ‘semantically transparent’ corpora more readily available for end-to-end NLG (see e.g., Chen & Mooney, 2008; Liang et al., 2009, and the discussion in Section 3.3).

7.1.3 Evaluating genre compatibility

A slightly different question that has occasionally been posed in evaluation studies asks whether the linguistic artefact produced by a system is a recognisable instance of a particular genre or style. We have seen examples of this in our discussion of creative language generation in Section 6. For example, one of the questions asked by Binsted et al. (1997) was whether the output of JAPE was recognisably a joke. Hardcastle and Scott (2008) describe an evaluation of a generation system for cryptic crossword clues based on a Turing test in which the objective was to determine whether the system’s outputs were recognisably different from human-authored clues.

While such questions clearly have an intrinsic orientation, they also have a bearing on extrinsic factors, since the ability to recognise an artefact as an instance of a genre or as exhibiting a certain style or personality is arguably one of the sources of its impact, especially in the case of creative language use.

Of course, the intention behind variation in style, personality or affect may well be to ultimately increase effectiveness in achieving some ulterior goal. Indeed, any NLG system intended to be embedded in a specific environment will need to address stylistic and genre-based issues. For example, our hypothetical weather report generator might use a very brief, technical style given its professional pool of target users (as was the case with SUMTIME Reiter et al., 2005); in contrast, weather reports intended for public consumption, such as those in

the WEATHERGOV corpus, would probably be longer and less technical (Angeli et al., 2010).

However, there is a difference between evaluating whether genre constraints or stylistic variation help contribute to a goal, and evaluating whether the text actually exhibits the desired variation. For example, Mairesse and Walker (2011) evaluated the PERSONAGE system (see Section 5) by asking users to judge personality traits as reflected in generated dialogue fragments (rather than, say, measuring whether users were more likely to eat at a restaurant if this was recommended by a configuration of the system with a high degree of extraversion). This is similar in spirit to the question about jokehood asked by Binsted et al. (1997), in contrast to the more explicitly extrinsic evaluation of the STANDUP joke generator by Waller et al. (2009), which asked whether the system actually helped users improve their interactions with peers.

7.2 Extrinsic evaluation methods

In contrast to intrinsic methods, extrinsic evaluations measuring effectiveness in achieving a desired goal. In the example scenario of Figure 8, such an evaluation might address the impact on planning by the engineers who are the target users of the system. Clearly, ‘effectiveness’ is dependent on the application domain and purpose of a system. Examples include:

- persuasion and behaviour change, for example, through exposure to personalised smoking cessation letters (Reiter et al., 2003);
- purchasing decision after presentation of arguments for and against options on the housing market based on a user model (Carenini & Moore, 2006);
- engagement with ecological issues after reading blogs about migrating birds (Siddharthan et al., 2013);
- decision support in a medical setting following the generation of patient reports (Portet et al., 2009; Hunter et al., 2012);
- enhancing linguistic interaction among users with complex communication needs via the generation of personal narratives (Tintarev et al., 2016);
- enhancing learning efficacy in tutorial dialogue (Di Eugenio et al., 2005; Fossati et al., 2015; Boyer et al., 2011; Lipschultz et al., 2011; Chi et al., 2014)

While questionnaire-based or self-report studies can be used to address extrinsic criteria (e.g., Hunter et al., 2012; Siddharthan et al., 2013; Carenini & Moore, 2006), in many cases evaluation relies on some objective measure of performance or achievement. This can be done with the target users IN SITU, enhancing the ecological validity of the study, but can also take the form of a task that models the scenarios for which the NLG system has been designed. Thus, in the GIVE Challenge (Striegnitz et al., 2011), in which NLG systems

generated instructions for a user to navigate through a virtual world, a large-scale task-based evaluation was carried out by having users play the GIVE game online, while various indices of success were logged, including the time it took a user to complete the game. REG algorithms whose goal was to generate identifying descriptions of objects in visual domains, were evaluated in part based on the time it took readers to identify a referent based on a generated description, as well as their error rate(Gatt & Belz, 2010). SKILLSUM, a system to generate feedback reports from literacy assessments, was evaluated by measuring how user's self-assessment of their own literacy skills improved after reading generated feedback, compared to control texts (Williams & Reiter, 2008).

A potential drawback of extrinsic studies, in addition to time and expense, is a reliance on an adequate user base (which can be difficult to obtain when users have to be sampled from a specific population, such as the engineers in our hypothetical scenario in Figure 8) and the possibility of carrying out the study in a realistic setting. Such studies also raise significant design challenges, due to the need to control for intervening and confounding variables, comparing multiple versions of a system (e.g. in an ablative design; see Section 7.3 below), or comparing a system against a gold standard or baseline. For example, Carenini and Moore (2006) note that evaluating the effectiveness of arguments presented in text needs to take into account aspects of a user's personality which may impact how receptive they are to arguments in the first place.

An example of the trade-off between design and control issues and ecological validity is provided by the BabyTalk family of systems. A pilot system called BT-45 (Portet et al., 2009), which generated patient summaries from 45-minute spans of historical patient data, was evaluated in a task involving nurses and doctors, whose chose from among a set of clinical actions to take based on the information given. These were then compared to 'ground truth' decisions by senior neonatal experts. This evaluation was carried out off-ward; hence, subjects took clinical decisions in an artificial environment without direct access to the patient. On the other hand, in the evaluation of BT-NURSE, a successor to BT-45 which summarised patient data collected over a twelve-hour shift (Hunter et al., 2012), the system was evaluated on-ward using live patient data, but ethical considerations precluded a task-based evaluation. For the same reasons, comparison to 'gold standard' human texts was also impossible. Hence, the evaluation elicited judgements, both on intrinsic criteria such as understandability and accuracy and on extrinsic criteria such as perceived clinical utility (see Siddharthan et al., 2013, for a similarly indirect extrinsic measure of impact, this time in an ecological setting).

7.3 Black box vs glass box evaluation

With the exception of evaluations of specific modules or algorithms, as in the case of REG or surface realisers, most of the evaluation studies discussed so far would be classified as 'black box' evaluations of 'end-to-end', or complete, NLG systems. In a 'glass box' evaluation, on the other hand, it is the contribution of individual components that is under scrutiny, ideally in a setup where versions of

a system with and without a component are evaluated in the same manner. Note that the distinction between black box and glass box evaluation is orthogonal to the question of which methods are used.

An excellent example of a glass-box evaluation is Callaway and Lester (2002), who used an ablative design, eliciting judgements of the quality of the output of their narrative generation system based on different configurations that omitted or included key components. In a related vein, Elliott and Keller (2013) compared image-to-text models that included fine-grained dependency representations of spatial as well as linguistic dependencies, to models with a coarser-grained image representation, finding an advantage for the former.

However, exhaustive component-wise comparisons are sometimes difficult to make and may result in a combinatorial explosion of configurations, with a concomitant reduction in data points collected per configuration (assuming subjects are limited and need to be divided among different conditions) and a reduction in statistical power. Alternatives do exist in the literature. Reiter et al. (2003) elicited judgements on weather forecasts using human and machine-generated texts, together with a ‘hybrid’ version where the content was selected by forecasters, but the language was automatically generated. This enabled a comparison of human and automatic content selection. Angeli et al. (2010) used corpus-based and subjective measures to assess linguistic quality, coupled with precision and recall-based measures to assess content determination of their statistical system against human-annotated texts. In BT-NURSE (Hunter et al., 2012), nurses were prompted for free text comments (in addition to answering a questionnaire targeting extrinsic dimensions), which were then manually annotated and analysed to determine which elements of the system were potentially problematic.

7.4 On the relationship between evaluation methods

To what extent are the plethora of methods surveyed – from extrinsic, task-oriented to intrinsic ones relying on automatic metrics or human judgements – actually related? It turns out that multiple evaluation methods seldom give converging verdicts on a system, or on the relative ranking of a set of systems under comparison.

7.4.1 Metrics versus human judgements

Although corpus-based metrics used in MT and summarisation are typically validated by demonstrating their correlation with human ratings, meta-evaluation studies in these fields have suggested that the correspondence is somewhat weak (e.g., Dorr et al., 2004; Callison-Burch et al., 2006; Caporaso et al., 2008). Similarly, shared task evaluations on referring expression generation showed that corpus-based, judgement-based and experimental or task-based methods frequently do not correlate Gatt and Belz (2010). In their recent review Bernardi et al. (2016) note a similar issue in image captioning system evaluation. Thus, Kulkarni et al. (2013) found that their image description system did not out-

perform two earlier methods (Farhadi et al., 2010; Yang et al., 2011) on BLEU scores; however, human judgements indicated the opposite trend, with readers preferring their system (similar observations are made by Kiros et al., 2014). Hodosh et al. (2013) compared the agreement (measured by Cohen’s κ) between human judgements and BLEU or ROUGE scores for retrieved captions, finding that outputs were not ranked similarly by humans and metrics, unless the retrieved captions were identical to the reference captions.

On occasion, the correlation between a metric and human judgements appears to differ across studies, suggesting that metric-based results are highly susceptible to variation due to generation algorithms and datasets. For instance, Konstas and Lapata (2013) (discussed in Section 3.3.4 above) find that on corpus-based metrics, the best-performing version of their model does not outperform that of Kim and Mooney (2010) on the ROBOCUP domain, or that of Angeli et al. (2010) on their weather corpus (WEATHERGOV), though it performs better than Angeli et al. (2010) on the noisier ATIS travel dataset. However, an evaluation of fluency and semantic correctness, based on human judgements, showed that the system outperformed, by a small margin, both Kim and Mooney (2010) and Angeli et al. (2010) on both measures in all domains with the exception of WEATHERGOV, where Angeli et al.’s system did marginally better.

In a related vein, Elliott and De Vries (2015) compare their image captioning system, based on visual dependency relations, to the Bidirectional RNN developed by Karpathy and Fei-Fei (2015), on two different datasets. The two systems were close to each other on the VLT2K dataset, but not on Pascal1k, a result that the authors claim is due to VLT2K containing more pictures involving actions. As for the relationship between metrics and human judgements, Elliott and Keller (2013) concluded that METEOR correlates better than BLEU (see Elliott & Keller, 2014, for a systematic comparison of automatic metrics in this domain), a finding also confirmed in their later work (Elliott & De Vries, 2015), as well as in the MS-COCO Evaluation Challenge, which found that METEOR was more robust. However, work by Kuznetsova et al. (2014) showed variable results; their highest-scoring method as judged by humans, involving tree composition, was ranked higher by BLEU than by METEOR. In the MS-COCO Evaluation Challenge, some systems outperformed a human-human upper bound when compared to reference texts using automatic metrics, but no system reached this level in an evaluation based on human judgements (see Bernardi et al., 2016, for further discussion).

Some studies have explicitly addressed the relationship between methods as a research question in its own right. An important contribution in this direction is the study by (Reiter & Belz, 2009), which addressed the validity of corpus-based metrics in relation to human judgements, within the domain of weather forecast generation (a similar study has recently been conducted on image captioning by Elliott & Keller, 2014). In a first experiment, focussing on linguistic quality, the authors found a high correlation between expert and non-expert readers’ judgements, but the correlation between human judgements and the automatic metrics varied considerably (from 0.3 to 0.87), depending on the version of the metric used and whether the reference texts were included in the comparison

by human judges. The second experiment evaluated both linguistic quality, by asking human judges to rate clarity/readability; and content determination, by eliciting judgements of accuracy/appropriateness (by comparing texts to the raw data). The automatic metrics correlated significantly with judgements of clarity, but far less with accuracy, suggesting that they were better at predicting the linguistic quality than correctness.

Other studies have yielded similarly inconsistent results. In a study on paraphrase generation, Stent et al. (2005) found that automatic metrics correlated highly with judgements of adequacy (roughly akin to accuracy), but not fluency. By contrast, Espinosa et al. (2010) found that automatic metrics such as NIST, METEOR and GTM correlate moderately well with human fluency and adequacy judgements of English surface realisation quality, while Cahill (2009) reported only a weak correlation for German surface realisation. Wubben et al. (2012), comparing text simplification strategies, found low, but significant correlations between BLEU and fluency judgements, and a very low, *negative* correlation between BLEU and adequacy. These contrasting findings suggest that the relationship between metrics may depend on purpose and genre of the text under consideration; for example, Reiter and Belz (2009) used weather reports, while Wubben et al. (2012) used Wikipedia articles.

Various factors can be adduced to explain the inconsistency of these meta-evaluation studies:

1. Metrics such as BLEU are sensitive to the length of the texts under comparison. With shorter texts, n-gram based metrics are likely to result in lower scores.
2. The type of overlap matters: for example, many evaluations in image captioning rely on BLEU-1 (the work of Elliott & Keller, 2013, 2014, was among the first to experiment with longer n-grams), but longer n-grams are harder to match, though they capture more syntactic information and are arguably better indicators of fluency.
3. Semantic variability is an important issue. Generated texts may be similar to reference texts, but differ on some near-synonyms, or subtle word order variations. As shown in Table 1, some metrics are designed to partially address these issues.
4. Many intrinsic corpus-based metrics are designed to compare against multiple reference texts, but this is not always possible in NLG. For example, while image captioning datasets typically contain multiple captions per image (typically, around 5), this is not the case in other domains, like weather reporting or restaurant recommendations.

The upshot is that NLG evaluations increasingly rely on multiple methods, a trend that is equally visible in other areas of NLP, such as MT (Callison-Burch et al., 2007, 2008).

7.4.2 Using controlled experiments

A few studies have validated evaluation measures against experimental data. For example, Siddharthan and Katsos (2012) compared the outcomes of their magnitude estimation judgement study (see Section 7.1 above) to the results from a sentence recall task, finding that the results from the latter are largely consistent with judgements and concluding that they can substitute for task-based evaluations to shed light on breakdowns in comprehension at sentence level. A handful of studies have also used behavioral experiments and compared ‘online’ processing measures, such as reading time of referring expressions, to corpus-based metrics (e.g. Belz et al., 2010). Correlations with automatic metrics are usually poor. A somewhat different use of reading times was made by Lapata (2006), who used them as an objective measure against which to validate Kendall’s τ as a metric for assessing information ordering in text (an aspect of text structuring). In a recent study, Zarrieß et al. (2015) compared generated texts to human-authored and ‘filler’ texts (which were manually manipulated to compromise their coherence). They found that reading-time measures were more useful to distinguish these classes of texts than offline measures based on elicited judgements of fluency and clarity.

7.5 Evaluation: Concluding remarks

Against the background of this section, three main conclusions can be drawn:

1. There is a widespread acceptance of the necessity of using multiple evaluation methods in NLG. While these are not always consistent among themselves, they are useful in shedding light on different aspects of quality, from fluency and clarity of output, to adequacy of semantic content and effectiveness in achieving communicative intentions. The choice of method has a direct impact on the way in which results can be interpreted.
2. Meta-evaluation studies have yielded conflicting results on the relationship between human judgements, behavioural measures and automatically computed metrics. The correlation among them varies depending on task and application domain. This is a subject of ongoing research, with plenty of studies focussing on the reliability of metrics and their relationship to other measures, especially human judgements.
3. A question that remains under-explored concerns the dimensions of quality that are themselves the object of inquiry. (In this connection, it is worth noting that some kindred disciplines have sought to de-emphasise their role on the grounds that they are inconsistent; see Callison-Burch et al. (2008), for example). For example, what are people judging when they judge fluency or adequacy and how consistently do they do so? It is far from obvious whether these judgements should really be expected to correlate with other measures, given that the latter are *producer*-oriented, focussing on output, while judgements are themselves often *receiver*-oriented, focussing on how the output is read or processed (for a related argument,

see Oberlander, 1998). Furthermore, while meta-linguistic judgements can be expected to reflect the impact of a text on its readers, there is nevertheless the possibility that behavioural, online methods designed to directly investigate aspects of processing would yield a different picture, a result that has been obtained in some psycholinguistic studies (e.g. Engelhardt et al., 2006).

In conclusion, our principal recommendation to NLG practitioners, where evaluation is concerned, is to err in favour of diversity, by using multiple methods, as far as possible, and reporting not only their results, but also the correlation between them. Weak correlations need not imply that the results of a particular method are invalid. Rather, they may indicate that measures focus on different aspects of a system or its output.

8 Discussion and future directions

Over the past two decades, the field of NLG has advanced considerably, and many of these recent advances have not been covered in a comprehensive survey yet. This paper has sought to address this gap, with the following goals:

1. to give an update of the core tasks and architectures in the field, with an emphasis on recent data-driven techniques;
2. to briefly highlight recent developments in relatively new areas, including vision-to-text generation and the generation of stylistically varied, engaging or creative texts; and
3. to extensively discuss the problems and prospects of evaluating NLG applications.

Throughout this survey, various general, related themes have emerged. Probably the central theme has been the gradual shift away from traditional, rule-based approaches to statistical, data-driven ones, which, of course, has been taking place in AI in general. In NLG, this has had substantial impact on how individual tasks are approached (e.g., moving away from domain-dependent to more general, domain-independent approaches, relying on available data instead) as well as on how tasks are combined in different architectures (e.g., moving away from modular towards more integrated approaches). The trade-off between output quality of the generated text and the efficiency and robustness of an approach is becoming a central issue: data-driven approaches are arguably more efficient than rule-based approaches, but the output quality may be compromised, for reasons we have discussed. Another important theme has been the increased interplay between core NLG research and other disciplines, such as computer vision (in the case of vision-to-text) and computational creativity research (in the case of creative language use).

At the conclusion of this comprehensive survey of the state of the art in NLG, and given the fast pace at which developments occur both in industry and

academia, we feel it is useful to point to some potential future directions, as well as to raise a number of questions which recent research has brought to the fore.

8.1 Why (and how) should NLG be used?

More than a decade ago, towards the beginning of their influential survey on NLG, Reiter and Dale (2000) recommended to the developer that she pose this question before embarking on the design and implementation of a system. Can NLG really help in the target domain? Does a cheaper, more standard solution exist and would it work just as well? From the perspective of an engineer or a company, these are obviously relevant questions. As recent industry-based applications of NLG show, this technology is typically valuable whenever information that needs to be presented to users is relatively voluminous, and comes in a form which is not easily consumed and does not afford a straightforward mapping to a more user-friendly modality without considerable transformation. This is arguably where NLG comes into its own, offering a battery of techniques to select, structure and present the information.

However, the question whether NLG is worth using in a specific setting should also be accompanied by the question of *how* it should be used. Our survey has focussed on techniques for the generation of text, but text is not always presented in isolation. Other important dimensions include document structure and layout, an under-studied problem (but see Power et al., 2003). They also include the role of graphics in text, an area where there is the potential for further interaction between the NLG and visualisation communities, addressing such questions as which information should be rendered textually and which can be made more accessible in a graphical modality. These questions are of great relevance in some domains, especially those where accurate information delivery is a precursor to decision-making in fault-critical situations (for some examples, see Elting et al., 1999; Law et al., 2005).

8.2 NLG isn't about text-to-text... or is it?

In our introductory section, we distinguished text-to-text generation from data-to-text generation; this survey has focussed primarily on the latter. The two areas have distinguishing characteristics, not least the fact that NLG inputs tend to vary widely, as do the goals of NLG systems as a function of the domain under consideration. In contrast, the input in text-to-text generation, especially Automatic Summarisation, is comparatively homogeneous, and while its goals can vary widely, the field has also been successful at defining tasks and datasets (for instance, through the DUC shared tasks), which have set the standard for subsequent research.

Yet, a closer look at the two types of generation will show more scope for convergence than the above characterisation suggests. To begin with, if NLG is concerned with going from data to text, then surely textual input should be considered as one out of broad variety of forms in which input data might

be presented. Some recent work, such as Kondadadi et al. (2013) (discussed in Section 3.3) and McIntyre and Lapata (2009) (discussed in Section 6) has explicitly focussed on leveraging such data to generate coherent text. Other approaches to NLG, including some systems that conform to a standard, modular, data-to-text architecture (e.g., Hunter et al., 2012), have had to deal with text as one out of a variety of input types, albeit using very simple techniques. Generation from heterogeneous inputs which include text as one type of data is a promising research direction, especially in view of the large quantities of textual data available, often accompanied by numbers or images.

8.3 Theories and models in search of applications?

In their overview of the status of evaluation in NLG in the late 1990s, Mellish and Dale (1998) discussed, among the possible ways of evaluating a system, its theoretical underpinnings and in particular whether the theoretical model underlying an NLG system or one of its components is adequate to the task and can generalise to new domains. Rather than evaluating an NLG system as such, this question targets the theory itself, and suggests that we view NLG as a potential testbed for such theories or models. But what are the theories that underlie NLG?

The prominence of theoretical models in NLG tends to depend on the task under consideration. For instance, many approaches to realisation discussed in Section 2.6 are based on a specific theory of syntactic structure; research on REG has often been based on insights from pragmatic theory, especially the Gricean maxims (Grice, 1975); and much research on text structuring has been inspired by Rhetorical Structure Theory (Mann & Thompson, 1988). Relatively novel takes on various sentence planning tasks – especially those concerned with style, affect and personality – tend to have a theoretical inspiration, in the form of a model of personality (John & Srivastava, 1999) or a theory of politeness (Brown & Levinson, 1987), for example.

More often than not, such theories are leveraged in the process of formalising a particular problem to achieve a tractable solution. Treating their implementation in an NLG system as an explicit test of the theory, as Mellish and Dale (1998) seem to suggest, happens far less often. This is perhaps a reflection of a division between ‘engineering-oriented’ and ‘theoretically-oriented’ perspectives in the field: the former perspective emphasises workable solutions, robustness and output quality; the latter emphasises theoretical soundness, cognitive plausibility and so forth. However, the theory/engineering dichotomy is arguably a false one. While the goal of NLG research is often different from, say, that of cognitive modelling (for example, few NLG systems seek to model production errors explicitly), it is also true that theory-driven implementations are themselves worthy contributions to theoretical work.

Recently, some authors have argued that NLG practitioners should pay closer attention to theoretical and cognitive models. The reasons marshalled in favour of this argument are twofold. First, psycholinguistic results and theoretical models can actually help to improve implemented systems, as Rajkumar and

White (2014) show for the case of realisation. Second, as argued for example by van Deemter et al. (2012b), theoretical models can benefit from the formal precision that is the bread-and-butter of computational linguistic research; a concrete case in point in NLP is provided by Poesio et al. (2004), whose implementation of Centering Theory (Grosz et al., 1995) shed light on a number of underspecified parameters in the original model and subsequent modifications of it. Our argument here is that NLG has provided a wealth of theoretical insights which should not be lost to the broader research community; similarly, NLG researchers would undoubtedly benefit from an awareness of recent developments in theoretical and experimental work.

8.4 Where do we go from here?

Finally, we conclude with some speculations on some further directions for future research for which the time seems ripe.

Within the field of Natural Language Processing as a whole, a remarkable recent development is the explosion of interest in social media, including online blogs, micro-blogs such as Twitter feeds, and social platforms such as Facebook. In one respect, interest in social media could be seen as a natural extension of long-standing topics in NLP, including the desire to deal with language ‘in the wild’. However, social media data has given more impetus to the exploration of non-canonical language (e.g. Eisenstein, 2013); the impact of social and demographic factors on language use (e.g. Hovy & Søgaard, 2015; Johannsen et al., 2015); the prevalence of paralinguistic features such as affect, irony and humour (Pang & Lee, 2008; Lukin & Walker, 2013); and other variables such as personality (e.g. Oberlander & Nowson, 2006; Farnadi et al., 2013; Schwartz et al., 2013). Social media feeds are also important data streams for the identification of topical and trending events (see Atefah & Khreich, 2015, for a recent review). There is as yet little work on generating textual or multimedia summaries of such data (but see, for example, Wang et al., 2014) or generating text in social media contexts (exceptions include Ritter et al., 2011; Cagan et al., 2014). Since much of social media text is subjective and opinionated, an increased interest in social media on the part of NLG researchers may also give new impetus to research on the impact of style, personality and affect on textual variation (discussed in Section 5), and on non-literal language (including some of the phenomena discussed in Section 6).

A second potential growth area for NLG is situated language generation. The term *situated* is usually taken to refer to language use in physical or virtual environments where production choices explicitly take into account perceptual and physical properties. Research on situated language processing has advanced significantly in the past several years, with frameworks for language production and understanding in virtual contexts (e.g., Kelleher et al., 2005), as well as a number of contributions within NLG, especially for the generation of language in interactive environments (Kelleher & Kruijff, 2006; Stoia & Shockley, 2006; Garoufi & Koller, 2013; Dethlefs & Cuayáhuitl, 2015). The popular GIVE Challenge added further impetus to this research (Striegnitz et al., 2011). Clearly,

this work is also linked to the enterprise of grounding generated language in the perceptual world, of which the research discussed in Section 4 constitutes one of the current trends. However, there are many fields where situatedness is key, in which NLG can still make novel contributions. One of these is gaming. With the exception of a few endeavours to enhance the variety of linguistic expressions used in virtual environments (e.g., Orkin & Roy, 2007), NLG technology is relatively unrepresented in research on games, despite significant progress on dynamic content generation in game environments (e.g., Togelius et al., 2011). This may be due to the perception that linguistic interaction in games is predictable and can rely on ‘canned’ text. However, with the growing influence of gamification as a strategy for enhancing a variety of activities beyond entertainment, such as pedagogy, as well as the development of sophisticated planning techniques for varying the way in which game worlds unfold on the fly, the assumption of predictability where language use is concerned may well be up for revision.

Third, there is a growing interest in applying NLG techniques to generation from structured knowledge bases and ontologies (e.g. Ell & Harth, 2014; Duma & Klein, 2013; Gyawali & Gardent, 2014; Mrabet et al., 2016; Sleimi & Gardent, 2016, some of which were briefly discussed in Section 3.3.4). The availability of knowledge bases such as DBPedia, or folksonomies such as Freebase, not only constitute input sources in their own right, but also open up the possibility of exploring alignments between structured inputs and text in a broader variety of domains than has hitherto been the case.

Finally, while there has been a significant shift in the past few years towards data-driven techniques in NLG, many of these have not been tested in commercial or real-world applications, despite the growth in commercialisation of bespoke text generation services noted in the introductory section. Typically, the arguments for rule-based systems in commercial scenarios, or in cases where input is high-volume and heterogeneous, are that (1) their output is easier to control for bespoke systems; or (2) that data is in any case unavailable in a given domain, rendering the use of statistical techniques moot; or (3) data-driven systems have not been shown to be able to scale up beyond experimental scenarios (some of these arguments are made, for instance, by Harris, 2008). A response to the first point depends on the availability of techniques which enable the developer to ‘look under the hood’ and understand the statistical relationships learned by a model. Such techniques are, for example, being developed to investigate or visualise the representations learned by deep neural networks. The second point calls for more investment in research on data acquisition and data-text alignment. Techniques for generation which rely on less precise alignments between data and text are also a promising future direction. Finally, scalability remains an open challenge. Many of the systems we have discussed have been developed within research environments, where the aim is of course to push the frontiers of NLG and demonstrate feasibility or correctness of novel approaches. While in some cases, research on data-to-text has addressed large-scale problems – notably in some of the systems that summarise numerical data – a greater concern with scalability would also focus researchers’ attention on issues such

as the time and resources required to collect data and train a system and the efficiency of the algorithms being deployed. Clearly, developments in hardware will alleviate these problems, as has happened with some statistical methods that have recently become more feasible.

9 Conclusion

Recent years have seen a marked increase in interest in automatic text generation. Companies now offer NLG technology for a range of applications in domains such as journalism, weather, and finance. The huge increase in available data and computing power, as well as rapid developments in machine-learning, have created many new possibilities and motivated NLG researchers to explore a number of new applications, related to, for instance, image-to-text generation, while applications related to social media seem to be just around the corner, as witness, for instance, the emergence of NLG-related techniques for automatic content-creation as well as NLG for twitter and chatbots (e.g., Dale, 2016). With developments occurring at a steady pace, and the technology also finding its way into industrial applications, the future of the field seems bright. In our view, research in NLG should be further strengthened by more collaboration with kindred disciplines. It is our hope that this survey will serve to highlight some of the potential avenues for such multi-disciplinary work.

Acknowledgements

This work has greatly benefitted from discussions with and comments from Grzegorz Chrupala, Robert Dale, Raquel Hervás, Thiago Castro Ferreira, Ehud Reiter, Marc Tanti, Mariët Theune, Kees van Deemter, Michael White and Sander Wubben. EK received support from RAAK-PRO SIA (2014-01-51PRO) and The Netherlands Organization for Scientific Research (NWO 360-89-050), which is gratefully acknowledged.

References

- Anderson, P., Fernando, B., Johnson, M., & Gould, S. (2016). SPICE: Semantic Propositional Image Caption Evaluation. In *Proc. ECCV'16*, pp. 1–17.
- Androutsopoulos, I., & Malakasiotis, P. (2010). A survey of paraphrasing and textual entailment methods. *Journal of Artificial Intelligence Research*, 38, 135–187.
- Angeli, G., Liang, P., & Klein, D. (2010). A Simple Domain-Independent Probabilistic Approach to Generation. In *Proc. EMNLP'10*, pp. 502–512.
- Angeli, G., Manning, C. D., & Jurafsky, D. (2012). Parsing time: Learning to interpret time expressions. In *Proc. NAACL-HLT'12*, pp. 446–455.

- Antol, S., Agrawal, A., Lu, J., Mitchell, M., Batra, D., Zitnick, C. L., & Parikh, D. (2015). VQA: Visual Question Answering. In *Proc. ICCV'15*, pp. 2425–2433.
- Antol, S., Zitnick, C. L., & Parikh, D. (2014). Zero-shot learning via visual abstraction. In *Proc. ECCV'14*, pp. 401–416.
- Appelt, D. (1985). *Planning English Sentences*. Cambridge University Press, Cambridge, UK.
- Argamon, S., Koppel, M., Pennebaker, J. W., & Schler, J. (2007). Mining the Blogosphere: Age, gender and the varieties of self-expression. *First Monday*, 12(9).
- Atefah, F., & Khreich, W. (2015). A survey of techniques for event detection in twitter. *Computational Intelligence*, 31(1), 132–164.
- Austin, J. L. (1962). *How to do things with words*. Clarendon Press, Oxford.
- Bahdanau, D., Cho, K., & Bengio, Y. (2015). Neural Machine Translation By Jointly Learning To Align and Translate. In *Proc. ICLR'15*, pp. 1–15.
- Bal, M. (2009). *Narratology* (Third edition). University of Toronto Press, Toronto.
- Ballesteros, M., Bohnet, B., Mille, S., & Wanner, L. (2015). Data-driven sentence generation with non-isomorphic trees. In *Proc. NAACL-HTL'15*, pp. 387–397.
- Banaee, H., Ahmed, M. U., & Loutfi, A. (2013). Towards NLG for Physiological Data Monitoring with Body Area Networks. In *Proc. ENLG'13*, pp. 193–197.
- Bangalore, S., & Rambow, O. (2000). Corpus-based lexical choice in Natural Language Generation. In *Proc. ACL'00*, pp. 464–471.
- Bangalore, S., & Stent, A. (2014). *Natural Language Generation in Interactive Systems*. Cambridge University Press.
- Banik, E., Gardent, C., & Kow, E. (2013). The KBGen Challenge. In *Proc. ENLG'13*, pp. 94–97.
- Bannard, C., & Callison-Burch, C. (2005). Paraphrasing with bilingual parallel corpora. In *Proc. ACL'05*, pp. 597–604.
- Bard, E. G., Robertson, D., & Sorace, A. (1996). Magnitude Estimation of Linguistic Acceptability. *Lamguage*, 72(1), 32–68.
- Barnard, K. (2016). *Computational Methods for Integrating Vision and Language*. Morgan and Claypool Publishers.
- Bartoli, A., De Lorenzo, A., Medvet, E., & Tarlao, F. (2016). Your paper has been accepted, rejected, or whatever: Automatic generation of scientific paper reviews. In *International Conference on Availability, Reliability, and Security*, pp. 19–28.

- Barzilay, R., Elhadad, N., & McKeown, K. R. (2002). Inferring strategies for sentence ordering in multidocument news summarization. *Journal of Artificial Intelligence Research*, 17, 35–55.
- Barzilay, R., & Lapata, M. (2005). Collective content selection for concept-to-text generation. In *Proc. HLT/EMNLP'05*, pp. 331–338.
- Barzilay, R., & Lapata, M. (2006). Aggregation via Set Partitioning for Natural Language Generation. In *Proc. HLT-NAAACL'06*, pp. 359–366.
- Barzilay, R., & Lee, L. (2004). Catching the Drift: Probabilistic Content Models, with Applications to Generation and Summarization. In *Proc. HLT-NAAACL'04*, pp. 113–120.
- Bateman, J. A. (1997). Enabling technology for multilingual natural language generation: the KPML development environment. *Natural Language Engineering*, 3(1), 15–55.
- Bateman, J. A., & Zock, M. (2005). Natural Language Generation. In Mitkov, R. (Ed.), *The Oxford Handbook of Computational Linguistics*. Oxford University Press, Oxford, UK.
- Belz, A. (2003). And Now with Feeling: Developments in Emotional Language Generation (Technical Report No. ITRI-03-21). Tech. rep., University of Brighton, Brighton, UK.
- Belz, A. (2008). Automatic generation of weather forecast texts using comprehensive probabilistic generation-space models. *Natural Language Engineering*, 14(04).
- Belz, A., & Kow, E. (2010). Comparing rating scales and preference judgements in language evaluation. In *Proc. INLG'10*, pp. 7–15.
- Belz, A., & Kow, E. (2011). Discrete vs . Continuous Rating Scales for Language Evaluation in NLP. In *Proc. ACL'11*, pp. 230–235.
- Belz, A., Kow, E., Viethen, J., & Gatt, A. (2010). Generating referring expressions in context: The GREC task evaluation challenges. In Krahmer, E., & Theune, M. (Eds.), *Empirical Methods in Natural Language Generation*. Springer, Berlin and Heidelberg.
- Belz, A., White, M., Espinosa, D., Kow, E., Hogan, D., & Stent, A. (2011). The First Surface Realisation Shared Task : Overview and Evaluation Results. In *Proc. ENLG'11*, Vol. 2, pp. 217–226.
- Bengio, Y., Ducharme, R., Vincent, P., & Janvin, C. (2003). A Neural Probabilistic Language Model. *Journal of Machine Learning Research*, 3, 1137–1155.
- 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.
- Biber, D. (1988). *Variation Across Speech and Writing*. Cambridge University Press, Cambridge.

- Binsted, K., Bergen, B., & McKay, J. (2003). Pun and non-pun humour in second-language learning. In *Proc. CHI 2003 Workshop on Humor Modeling in the Interface*.
- Binsted, K., Pain, H., & Ritchie, G. D. (1997). Children's evaluation of computer-generated punning riddles. *Pragmatics & Cognition*, 5(2), 305–354.
- Binsted, K., & Ritchie, G. D. (1994). An implemented model of punning riddles. In *Proc. AAAI'94*.
- Binsted, K., & Ritchie, G. D. (1997). Computational rules for generating punning riddles. *Humor: International Journal of Humor Research*, 10(1), 25–76.
- Bohnet, B. (2008). The fingerprint of human referring expressions and their surface realization with graph transducers. In *Proc. INLG'08*, pp. 207–210.
- Bohnet, B., Wanner, L., Mille, S., & Burga, A. (2010). Broad Coverage Multilingual Deep Sentence Generation with a Stochastic Multi-Level Realizer. In *Proc. COLING'10*, pp. 98–106.
- Bollegrala, D., Okazaki, N., & Ishizuka, M. (2010). A bottom-up approach to sentence ordering for multi-document summarization. *Information Processing & Management*, 46(1), 89–109.
- Bollmann, M. (2011). Adapting SimpleNLG for German. In *Proc. ENLG'11*.
- Bouayad-Agha, N., Casamayor, G., Wanner, L., & Mellish, C. (2013). Overview of the First Content Selection Challenge from Open Semantic Web Data. In *Proc. ENLG'11*, pp. 98–102.
- Boyer, K. E., Phillips, R., Ingram, A., Ha, E. Y., Wallis, M., Vouk, M., & Lester, J. C. (2011). Investigating the relationship between dialogue structure and tutoring effectiveness: A hidden markov modeling approach. *International Journal of Artificial Intelligence in Education*, 21(1-2), 65–81.
- Brants, T., & Franz, A. (2006). Web 1T 5-gram Version 1. Tech. rep., Linguistic Data Consortium.
- Bratman, M. E. (1987). *Intentions, Plans and Practical Reason*. CSLI, Stanford, CA.
- Bringsjord, S., & Ferrucci, D. A. (1999). *Artificial Intelligence and Literary Creativity: Inside the Mind of BRUTUS, a Storytelling Machine*. Lawrence Erlbaum Associates, Hillsdale, NJ.
- Brown, J. C., Frishkoff, G. A., & Eskenazi, M. (2005). Automatic question generation for vocabulary assessment. In *Proc. EMNLP'05*, pp. 819–826.
- Brown, P., & Levinson, S. C. (1987). *Politeness: Some Universals in Language Usage*. Cambridge University Press, Cambridge, UK.
- Bruner, J. (2011). The Narrative Construction of Reality. *Critical Inquiry*, 18(1), 1–21.

- Busemann, S., & Horacek, H. (1997). Generating Air Quality Reports From Environmental Data. In Busemann, S., Becker, T., & Finkler, W. (Eds.), *DFKI Workshop on Natural Language Generation (DFKI Document D-97-06)*, pp. 1–7. DFKI, Saarbrücken.
- Cagan, T., Frank, S. L., & Tsarfaty, R. (2014). Generating Subjective Responses to Opinionated Articles in Social Media: An Agenda-Driven Architecture and a Turing-Like Test. In *Proc. Joint Workshop on Social Dynamics and Personal Attributes in Social Media*, pp. 58–67.
- Cahill, A. (2009). Correlating Human and Automatic Evaluation of a German Surface Realiser. In *Proc. ACL-IJCNLP'09*, pp. 97–100.
- Cahill, A., Forst, M., & Rohrer, C. (2007). Stochastic realisation ranking for a free word order language. In *Proc. ENLG'07*, pp. 17–24.
- Cahill, A., & Josef, V. (2006). Robust PCFG-Based Generation using Automatically Acquired LFG Approximations. In *Proc. COLING-ACL'06*, pp. 1033–1040.
- Callaway, C. B. (2005). The Types and Distributions of Errors in a Wide Coverage Surface Realizer Evaluation. In *Proc. ENLG'05*, pp. 162–167.
- Callaway, C. B., & Lester, J. C. (2002). Narrative prose generation. *Artificial Intelligence*, 139(2), 213–252.
- Callison-Burch, C., Fordyce, C., Koehn, P., Monz, C., & Schroeder, J. (2007). (Meta-) evaluation of machine translation. In *Proc. StatMT'07*, pp. 136–158.
- Callison-Burch, C., Fordyce, C., Koehn, P., Monz, C., & Schroeder, J. (2008). Further Meta-Evaluation of Machine Translation. In *Proc. StatMT'08*, pp. 70–106.
- Callison-Burch, C., Osborne, M., & Koehn, P. (2006). Re-evaluating the Role of BLEU in Machine Translation Research. In *Proc. EACL'06*.
- Caporaso, J. G., Deshpande, N., Fink, J. L., Bourne, P. E., Bretonnel Cohen, K., & Hunter, L. (2008). Intrinsic evaluation of text mining tools may not predict performance on realistic tasks. *Pacific Symposium on Biocomputing*, 13, 640–651.
- Carenini, G., & Moore, J. D. (2006). Generating and evaluating evaluative arguments. *Artificial Intelligence*, 170(11), 925–952.
- Carroll, J., & Oepen, S. (2005). High efficiency realization for a wide-coverage unification grammar. In Dale, R. (Ed.), *Proceedings of the 2nd International Joint Conference on Natural Language Processing (IJCNLP'05)*, pp. 165–176. Springer.
- Chang, F., Dell, G. S., & Bock, K. (2006). Becoming syntactic. *Psychological review*, 113(2), 234–72.
- Chen, D. L., & Mooney, R. J. (2008). Learning to sportscast: a test of grounded language acquisition. In *Proc. ICML'08*, pp. 128–135.

- Cheng, H., & Mellish, C. (2000). Capturing the interaction between aggregation and text planning in two generation systems. In *Proc. INLG '00*, Vol. 14, pp. 186–193.
- Chi, M., Jordan, P. W., & VanLehn, K. (2014). When Is Tutorial Dialogue More Effective Than Step-Based Tutoring?. In *Proc. ITS'14*, pp. 210–219.
- Clark, H. H. (1996). *Using Language*. Cambridge University Press, Cambridge, UK.
- Clarke, J., & Lapata, M. (2010). Discourse Constraints for Document Compression. *Computational Linguistics*, 36(3), 411–441.
- Clerwall, C. (2014). Enter the Robot Journalist. *Journalism Practice*, 8(5), 519–531.
- Coch, J. (1998). Interactive generation and knowledge administration in MultiMeteo. In *Proc. IWNLG'98*, pp. 300–303.
- Cohen, P. R., & Levesque, H. J. (1985). Speech acts and rationality. In *Proc. ACL'85*, pp. 49–60.
- Cohen, P. R., & Perrault, C. R. (1979). Elements of a plan-based theory of speech acts. *Cognitive Science*, 3, 177–212.
- Colin, E., Gardent, C., Mrabet, Y., Narayan, S., & Perez-Beltrachini, L. (2016). The webnlg challenge: Generating text from dbpedia data. In *Proc. INLG'16*, pp. 163–167, Edinburgh, UK.
- Colton, S., Goodwin, J., & Veale, T. (2012). Full-FACE Poetry Generation. In *Proc. ICCC'12*, pp. 95–102.
- Concepción, E., Méndez, G., Gervás, P., & León, C. (2016). A challenge proposal for narrative generation using cnls. In *Proc. INLG'16*, pp. 171–173, Edinburgh, UK.
- Cuayáhuitl, H., & Dethlefs, N. (2011). Hierarchical Reinforcement Learning and Hidden Markov Models for Task-Oriented Natural Language Generation. In *Proc. ACL'11*, pp. 654–659.
- Dale, R. (1989). Cooking up referring expressions. In *Proc. ACL'89*, pp. 68–75.
- Dale, R. (1992). *Generating Referring Expressions: Constructing Descriptions in a Domain of Objects and Processes*. MIT Press, Cambridge, MA.
- Dale, R. (2016). The return of the chatbots. *Natural Language Engineering*, 22(5), 811817.
- Dale, R., Anisimoff, I., & Narroway, G. (2012). Hoo 2012: A report on the preposition and determiner error correction shared task. In *Proc. 7th Workshop on Building Educational Applications Using NLP*, pp. 54–62.
- 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., & White, M. (2007). Shared Tasks and Comparative Evaluation in Natural Language Generation: Workshop Report. Tech. rep., Ohio State University, Arlington, Virginia.
- Dalianis, H. (1999). Aggregation in Natural Language Generation. *Computational Intelligence*, 15(4), 384–414.
- de Oliveira, R., & Sripada, S. (2014). Adapting SimpleNLG for Brazilian Portuguese realisation. In *Proc. INLG'14*, pp. 93–94.
- De Rosis, F., & Grasso, F. (2000). Affective Natural Language Generation. In Paiva, A. (Ed.), *Affective interactions*, pp. 204–218. Springer, Berlin and Heidelberg.
- De Smedt, K., Horacek, H., & Zock, M. (1996). Architectures for Natural Language Generation : Problems and Perspectives. In Adorni, G., & Zock, M. (Eds.), *Trends in Natural Language Generation: an Artificial Intelligence Perspective*, pp. 17–46. Springer, Berlin and Heidelberg.
- Dethlefs, N. (2014). Context-Sensitive Natural Language Generation : From Knowledge-Driven to Data-Driven Techniques. *Language and Linguistics Compass*, 8(3), 99–115.
- Dethlefs, N., & Cuayáhuitl, H. (2015). Hierarchical reinforcement learning for situated natural language generation. *Natural Language Engineering*, 21(3), 391–435.
- Devlin, J., Cheng, H., Fang, H., Gupta, S., Deng, L., He, X., Zweig, G., & Mitchell, M. (2015a). Language Models for Image Captioning : The Quirks and What Works. In *Proc. ACL/IJCNLP'15*, pp. 100–105.
- Devlin, J., Gupta, S., Girshick, R., Mitchell, M., & Zitnick, C. L. (2015b). Exploring Nearest Neighbor Approaches for Image Captioning. *Computing Research Repository (arXiv)*, 1505.04467, 1–6.
- Di Eugenio, B., Fossati, D., Yu, D., Haller, S., & Glass, M. (2005). Aggregation improves learning: Experiments in natural language generation for intelligent tutoring systems. In *Proc. ACL'05*, pp. 50–57, Ann Arbor, Michigan.
- Di Eugenio, B., & Green, N. (2010). Emerging applications of natural language generation in information visualization, education, and health-care. In Indurkhya, N., & Damerau, F. (Eds.), *Handbook of Natural Language Processing* (2 edition), p. 557575. Chapman and Hall/CRC, London.
- Di Fabbrizio, G., Stent, A., & Bangalore, S. (2008). Trainable Speaker-Based Referring Expression Generation. In *Proc. CoNLL'08*, pp. 151–158.
- DiMarco, C., Covvey, H. D., Bray, P., Cowan, D., DiCiccio, V., Hovy, E. H., Mulholland, D., & Lipa, J. (2007). The Development of a Natural Language Generation System For Personalized e-Health Information. In *Proc. MedInfo'07*.
- DiMarco, C., & Hirst, G. (1993). A Computational Theory of Goal-Directed Style in Syntax. *Computational Linguistics*, 19(3), 451–499.

- Dimitromanolaki, A., & Androutsopoulos, I. (2003). Learning to Order Facts for Discourse Planning in Natural Language Generation. In *Proc. ENLG'03*, pp. 23–30.
- Doddington, G. (2002). Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In *Proc. ARPA Workshop on Human Language Technology*, pp. 128–132.
- Donahue, J., Hendricks, L. A., Rohrbach, M., Venugopalan, S., Guadarrama, S., Saenko, K., & Darrell, T. (2015). Long-term Recurrent Convolutional Networks for Visual Recognition and Description. In *Proc. CVPR'15*, pp. 1–14.
- Dong, D., Wu, H., He, W., Yu, D., & Wang, H. (2015). Multi-Task Learning for Multiple Language Translation. In *Proc. ACL/IJCNLP'15*, pp. 1723–1732.
- Dorr, B., & Gaasterland, T. (1995). Selecting tense, aspect and connecting words in language generation. In *Proc. IJCAI'95*, pp. 1299–1305.
- Dorr, B., Monz, C., Oard, D., President, S., Zajic, D., & Schwartz, R. (2004). Extrinsic Evaluation of Automatic Metrics (LAMP-TR-115). Tech. rep., University of Maryland, College Park, MD.
- Dras, M. (2015). Evaluating human pairwise preference judgments. *Computational Linguistics*, 41(2), 309–317.
- Duboue, P. A., & McKeown, K. R. (2003). Statistical acquisition of content selection rules for natural language generation. In *Proc. EMNLP'03*, pp. 121–128.
- 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'13)*, pp. 83–94.
- Duygulu, P., Barnard, K., de Freitas, N., & Forsyth, D. (2002). Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In *Proc. ECCV'02*, pp. 97–112. Springer.
- Edmonds, P., & Hirst, G. (2002). Near-Synonymy and Lexical Choice. *Computational Linguistics*, 28(2), 105–144.
- Eisenstein, J. (2013). What to do about bad language on the internet. In *Proc. NAACL-HLT'13*, pp. 359–369.
- Elhadad, M., & Robin, J. (1996). An overview of SURGE: A reusable comprehensive syntactic realization component. In *Proceedings of the 8th International Natural Language Generation Workshop (IWNLG'98)*, pp. 1–4.
- Elhadad, M., Robin, J., & McKeown, K. R. (1997). Floating constraints in lexical choice. *Computational Linguistics*, 23(2), 195–239.
- Elhoseiny, M., Elgammal, A., & Saleh, B. (2017). Write a Classifier: Predicting Visual Classifiers from Unstructured Text Descriptions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, to appear, 1–14.

- Ell, B., & Harth, A. (2014). A language-independent method for the extraction of RDF verbalization templates. In *Proc. INLG'14*, pp. 26–34.
- Elliott, D., & De Vries, A. P. (2015). Describing Images using Inferred Visual Dependency Representations. In *Proc. ACL-IJCNLP'15*, pp. 42–52.
- Elliott, D., Frank, S., Sima'an, K., & Specia, L. (2016). Multi30K: Multilingual English-German Image Descriptions. *CoRR*, *abs/1605.00459*.
- Elliott, D., & Keller, F. (2013). Image Description using Visual Dependency Representations. In *Proc. EMNLP'13*, pp. 1292–1302.
- Elliott, D., & Keller, F. (2014). Comparing Automatic Evaluation Measures for Image Description. In *Proc. Volume 2: Short Papers*, pp. 452–457.
- Elman, J. L. (1990). Finding structure in time. *Cognitive Science*, *14*(2), 179–211.
- Elman, J. L. (1993). Learning and development in neural networks: The importance of starting small. *Cognition*, *48*, 71–99.
- Elson, D., & McKeown, K. R. (2010). Tense and aspect assignment in narrative discourse. In *Proc. INLG'10*, pp. 47–56.
- Elting, L. S., Martin, C. G., Cantor, S. B., & Rubenstein, E. B. (1999). Influence of data display formats on physician investigators' decisions to stop clinical trials: prospective trial with repeated measures. *BMJ (Clinical research ed.)*, *318*(7197), 1527–1531.
- Engelhardt, P., Bailey, K., & Ferreira, F. (2006). Do speakers and listeners observe the Gricean Maxim of Quantity?. *Journal of Memory and Language*, *54*(4), 554–573.
- Engonopoulos, N., & Koller, A. (2014). Generating effective referring expressions using charts. In *Proc. INLG'14*, pp. 6–15.
- Espinosa, D., Rajkumar, R., White, M., & Berleant, S. (2010). Further Meta-Evaluation of Broad-Coverage Surface Realization. In *Proc. EMNLP'10*, pp. 564–574.
- Espinosa, D., White, M., & Mehay, D. (2008). Hypertagging: Supertagging for surface realization with CCG. In *Proc. ACL-HLT'08*, pp. 183–191, Columbus, Ohio.
- Evans, R., Piwek, P., & Cahill, L. (2002). What is nlg?. In *Proc. INLG'02*, pp. 144–151.
- Fang, H., Gupta, S., Iandola, F., Srivastava, R., Deng, L., Dollár, P., Gao, J., He, X., Mitchell, M., Platt, J. C., Zitnick, C. L., & Zweig, G. (2015). From Captions to Visual Concepts and Back. In *Proc. CVPR'15*.
- Farhadi, A., Hejrati, M., Sadeghi, M. A., Young, P., Rashtchian, C., Hockenmaier, J., & Forsyth, D. (2010). Every picture tells a story: Generating sentences from images. In *Proc. ECCV'10*, Vol. 6314 LNCS, pp. 15–29.

- Farnadi, G., Zoghbi, S., Moens, M.-F., & De Cock, M. (2013). Recognising Personality Traits Using Facebook Status Updates. In *AAAI Technical Report WS-13-01: Computational Personality Recognition (Shared Task)*, pp. 14–18.
- Fass, D. (1991). met*: A Method for Discriminating Metonymy and Metaphor by Computer. *Computational Linguistics*, 17(1), 49–90.
- Feng, Y., & Lapata, M. (2010). How many words is a picture worth? Automatic caption generation for news images. In *Proc. ACL'10*, pp. 1239–1249.
- Ferraro, F., Mostafazadeh, N., Huang, T.-H., Vanderwende, L., Devlin, J., Galley, M., & Mitchell, M. (2015). A Survey of Current Datasets for Vision and Language Research. In *Proc. EMNLP'15*, pp. 207–213.
- Ferreira, T. C., Krahmer, E., & Wubben, S. (2016). Towards more variation in text generation: Developing and evaluating variation models for choice of referential form. In *Proc. ACL'16*, pp. 568 – 577.
- Ferreira, T. C., Krahmer, E., & Wubben, S. (2017). Generating flexible proper name references in text: Data, models and evaluation. In *Proc. EACL'17*.
- Fikes, R. E., & Nilsson, N. J. (1971). Strips: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence*, 2(3-4), 189–208.
- Filippova, K., & Strube, M. (2007). Generating Constituent Order in German Clauses. In *Proc. ACL'07*, pp. 320–327.
- Filippova, K., & Strube, M. (2009). Tree linearization in English: Improving language model based approaches. In *Proc. NAACL-HLT'09*, pp. 225–228.
- FitzGerald, N., Artzi, Y., & Zettlemoyer, L. (2013). Learning Distributions over Logical Forms for Referring Expression Generation. In *Proc. EMNLP'13*, pp. 1914–1925.
- Fleischman, M., & Hovy, E. H. (2002). Emotional Variation in speech-based Natural Language Generation. In *Proc. INLG'02*, pp. 57–64.
- Flower, L., & Hayes, J. R. (1981). A cognitive process theory of writing. *College composition and communication*, 32(4), 365–387.
- Fort, K., Adda, G., & Bretonnel Cohen, K. (2011). Amazon Mechanical Turk: Gold Mine or Coal Mine?. *Computational Linguistics*, 37(2), 413–420.
- Fossati, D., Di Eugenio, B., Ohlsson, S., Brown, C., & Chen, L. (2015). Data Driven Automatic Feedback Generation in the iList Intelligent Tutoring System. *Technology, Instruction, Cognition and Learning*, 10, 5–26.
- Frank, M. C., Goodman, N. D., & Tenenbaum, J. B. (2009). Using speakers’ referential intentions to model early cross-situational word learning. *Psychological Science*, 20(5), 578–85.
- Gardent, C. (2002). Generating Minimal Definite Descriptions. In *Proc. ACL'02*, pp. 96–103.

- Gardent, C., & Narayan, S. (2015). Multiple adjunction in feature-based tree-adjoining grammar. *Computational Linguistics*, 41(1), 41–70.
- Garoufi, K. (2014). PlanningBased Models of Natural Language Generation. *Language and Linguistics Compass*, 8(1), 1–10.
- Garoufi, K., & Koller, A. (2013). Generation of effective referring expressions in situated context. *Language and Cognitive Processes*, 00(00), 1–16.
- Gatt, A., & Belz, A. (2010). Introducing shared task evaluation to NLG: The TUNA shared task evaluation challenges. In Krahmer, E., & Theune, M. (Eds.), *Empirical methods in natural language generation*. Springer, Berlin and Heidelberg.
- Gatt, A., Portet, F., Reiter, E., Hunter, J. R., Mahamood, S., Moncur, W., & Sripana, S. (2009). From data to text in the neonatal intensive care Unit: Using NLG technology for decision support and information management. *AI Communications*, 22(3), 153–186.
- Gatt, A., van der Sluis, I., & van Deemter, K. (2007). Evaluating algorithms for the Generation of Referring Expressions using a balanced corpus. In *Proc. ENLG'07*, pp. 49–56.
- Geman, D., Geman, S., Hallonquist, N., & Younes, L. (2015). Visual Turing test for computer vision systems. *Proceedings of the National Academy of Sciences of the United States of America*, 112(12), 3618–23.
- Genette, G. (1980). *Narrative Discourse: An Essay in Method*. Cornell University Press, Ithaca, NY.
- Gervás, P. (2001). An expert system for the composition of formal Spanish poetry. *Knowledge-Based Systems*, 14(3-4), 181–188.
- Gervás, P. (2009). Computational approaches to storytelling and creativity. *AI Magazine*, Fall 2009, 49–62.
- Gervás, P. (2010). Engineering Linguistic Creativity: Bird Flight and Jet Planes. In *Proc. 2nd Workshop on Computational Approaches to Linguistic Creativity*, pp. 23–30.
- Gervás, P. (2012). From the Fleece of Fact to Narrative Yarns : a Computational Model of Composition. In *Proc. Workshop on Computational Models of Narrative*.
- Gervás, P. (2013). Story Generator Algorithms. In Hühn, P. (Ed.), *The Living Handbook of Narratology*. Hamburg University, Hamburg.
- Gkatzia, D., Rieser, V., Bartie, P., & Mackaness, W. (2015). From the Virtual to the Real World : Referring to Objects in Real-World Spatial Scenes. In *Proc. EMNLP'15*, pp. 1936–1942.
- Glucksberg, S. (2001). *Understanding figurative language: From metaphors to idioms*. Oxford University Press, Oxford.
- Godwin, K., & Piwek, P. (2016). Collecting Reliable Human Judgements on Machine-Generated Language : The Case of the QGSTEC Data . In *Proc. INLG'16*, pp. 212–216.

- Goldberg, E., Driedger, N., & Kittredge, R. I. (1994). Using Natural Language Processing to Produce Weather Forecasts. *IEEE Expert*, 2, 45–53.
- Goldberg, Y. (2016). A Primer on Neural Network Models for Natural Language Processing. *Journal of Artificial Intelligence Research*, 57, 345–420.
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning. Book in preparation for MIT Press.
- Goodman, J., Cryder, C., & Cheema, A. (2013). Data collection in a flat world: The strengths and weaknesses of mechanical turk samples. *Journal of Behavioral Decision Making*, 26(3), 213–224.
- Goyal, R., Dymetman, M., & Gaussier, E. (2016). Natural Language Generation through Character-Based RNNs with Finite-State Prior Knowledge. In *Proc. COLING'16*, pp. 1083–1092.
- Greene, E., Ave, L., Knight, K., & Rey, M. (2010). Automatic Analysis of Rhythmic Poetry with Applications to Generation and Translation. In *Proc. EMNLP'10*, pp. 524–533.
- Grice, H. P. (1975). Logic and conversation. In *Syntax and Semantics 3: Speech Acts*, pp. 41–58. Elsevier, Amsterdam.
- Grosz, B. J., Joshi, A. K., & Weinstein, S. (1995). Centering : A Framework for Modeling the Local Coherence of Discourse. *Computational Linguistics*, 21(2), 203–225.
- Guhe, M. (2007). *Incremental Conceptualization for Language Production*. Lawrence Erlbaum Associates, Hillsdale, NJ.
- Gupta, A., Verma, Y., & Jawahar, C. V. (2012). Choosing Linguistics over Vision to Describe Images. In *Proc. AAAI'12*, pp. 606–612.
- Gupta, S., Walker, M. A., & Romano, D. M. (2007). Generating Politeness in Task Based Interaction : An Evaluation of Linguistic Form and Culture. In *Proc. ENLG'07*, pp. 57–64.
- Gupta, S., Walker, M. A., & Romano, D. M. (2008). POLLY: A Conversational System that uses a Shared Representation to Generate Action and Social Language. In *Proc. IJCNLP'08*, pp. 7–12.
- Gyawali, B., & Gardent, C. (2014). Surface Realisation from Knowledge-Bases. In *Proc. ACL'14*, pp. 424–434.
- Halliday, M., & Matthiessen, C. M. (2004). *Introduction to Functional Grammar* (3rd Edition edition). Hodder Arnold, London.
- Harbusch, K., & Kempen, G. (2009). Generating clausal coordinate ellipsis multilingually: A uniform approach based on postediting. In *Proc. ENLG'09*, pp. 138–145.
- Hardcastle, D., & Scott, D. (2008). Can we evaluate the quality of generated text?. In *Proc. LREC'08*, pp. 3151–3158.
- Harnad, S. (1990). The symbol grounding problem. *Physica*, D42(1990), 335–346.

- Harris, M. D. (2008). Building a large-scale commercial NLG system for an EMR. In *Proc. INLG '08*, pp. 157–160.
- Hearst, M. A. (1992). Automatic Acquisition of Hyponyms from Large Text Corpora. In *Proc. COLING'92*, pp. 539–545.
- Heeman, P. A., & Hirst, G. (1995). Collaborating on referring expressions. *Computational Linguistics*, 21(3), 351–382.
- Hendricks, L. A., Akata, Z., Rohrbach, M., Donahue, J., Schiele, B., & Darrell, T. (2016a). Generating Visual Explanations. In *Proc. ECCV'16*.
- Hendricks, L. A., Venugopalan, S., Rohrbach, M., Mooney, R. J., Saenko, K., & Darrell, T. (2016b). Deep Compositional Captioning: Describing Novel Object Categories without Paired Training Data. In *Proc. CVPR'16*.
- Herman, D. (1997). Scripts, sequences and stories: Elements of a postclassical narratology. *PMLA*, 112(5), 1046–1059.
- Herman, D. (2001). Story logic in conversational and literary narratives. *Narrative*, 9(2), 130–137.
- Herman, D. (2007). Storytelling and the sciences of mind: Cognitive narratology, discursive psychology, and narratives in face-to-face interaction. *Narrative*, 15(3), 306–334.
- Hermida, A. (2015). From Mr and Mrs Outlier to Central Tendencies: Computational Journalism and crime reporting at the Los Angeles Times. *Digital Journalism*, 3(3), 381–397.
- Hervás, R., Arroyo, J., Francisco, V., Peinado, F., & Gervás, P. (2016). Influence of personal choices on lexical variability in referring expressions. *Natural Language Engineering*, 22(2), 257–290.
- Hervás, R., Francisco, V., & Gervás, P. (2013). Assessing the influence of personal preferences on the choice of vocabulary for natural language generation. *Information Processing & Management*, 49(4), 817–832.
- Hervás, R., Pereira, F., Gervás, P., & Cardoso, A. (2006). Cross-domain analogy in automated text generation. In *Proc. 3rd joint workshop on Computational Creativity*, Vol. 6.
- Hockenmaier, J., & Steedman, M. (2007). CCGbank: A Corpus of CCG Derivations and Dependency Structures Extracted from the Penn Treebank. *Computational Linguistics*, 33(3), 355–396.
- Hodosh, M., Young, P., & Hockenmaier, J. (2013). Framing image description as a ranking task: Data, models and evaluation metrics. *Journal of Artificial Intelligence Research*, 47, 853–899.
- Horacek, H. (1997). An Algorithm For Generating Referential Descriptions With Flexible Interfaces. In *Proc. ACL'97*, pp. 206–213.
- Hovy, D., & Søgaard, A. (2015). Tagging Performance Correlates with Author Age. In *ACL'15*, pp. 483–488.

- Hovy, E. H. (1988). *Generating Natural Language Under Pragmatic Constraints*. Lawrence Erlbaum Associates, Hillsdale, NJ.
- Hovy, E. H. (1991). Approaches to the Planning of Coherent Text. In Paris, C. L., Swartout, W. R., & Mann, W. C. (Eds.), *Natural Language Generation in Artificial Intelligence and Computational Linguistics*, pp. 83–102. Kluwer, Dordrecht.
- Hovy, E. H. (1993). Automated Discourse Planning and Generation. In *Proc. Annual Meeting of the Society for Text and Discourse*.
- Huang, T.-H., Ferraro, F., Mostafazadeh, N., Misra, I., Agrawal, A., Devlin, J., Girshick, R., He, X., Kohli, P., Batra, D., Zitnick, C. L., Parikh, D., Vanderwende, L., Galley, M., & Mitchell, M. (2016). Visual Storytelling. In *Proc. NAACL-HLT'16*, pp. 1233–1239.
- Hueske-Kraus, D. (2003). Suregen-2 : a shell system for the generation of clinical documents. In *Proc. EACL'03*, pp. 215–218.
- Hunter, J. R., Freer, Y., Gatt, A., Reiter, E., Sripada, S., & Sykes, C. (2012). Automatic generation of natural language nursing shift summaries in neonatal intensive care: BT-Nurse. *Artificial Intelligence in Medicine*, 56(3), 157–172.
- Hüske-Kraus, D. (2003). Text generation in clinical medicine: A review. *Methods of information in medicine*, 42(1), 51–60.
- Inui, K., Tokunaga, T., & Tanaka, H. (1992). Text revision: A model and its implementation. In Dale, R., Hovy, E. H., Rosner, D., & Stock, O. (Eds.), *Aspects of automated natural language generation*, Vol. 587, pp. 215–230. Springer, Berlin and Heidelberg.
- Janarthanam, S., & Lemon, O. (2011). The GRUVE Challenge: Generating Routes under Uncertainty in Virtual Environments. In *Proc. ENLG'11*, pp. 208–211.
- Janarthanam, S., & Lemon, O. (2014). Adaptive Generation in Dialogue Systems Using Dynamic User Modeling. *Computational Linguistics*, 40(4), 883–920.
- Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., & Darrell, T. (2014). Caffe: Convolutional Architecture for Fast Feature Embedding. In *Proc. ACM International Conference on Multimedia*, pp. 675–678. ACM.
- Johannsen, A., Hovy, D., & Søgaard, A. (2015). Cross-lingual syntactic variation over age and gender. In *Proc. CoNLL'15*, pp. 103–112.
- John, O., & Srivastava, S. (1999). The Big Five trait taxonomy: History, measurement, and theoretical perspectives. In Pervin, L., & John, O. (Eds.), *Handbook of Personality Theory and Research*. Guilford Press, New York.
- Johnson, J., Karpathy, A., & Fei-Fei, L. (2016). DenseCap: Fully Convolutional Localization Networks for Dense Captioning. In *Proc. CVPR'16*.

- Johnson, W. L., Rizzo, P., Bosma, W., Kole, S., Ghijsen, M., & Van Welbergen, H. (2004). Generating socially appropriate tutorial dialog. In Andre, E., Dybkjæ r, L., Minker, W., & Heisterkamp, P. (Eds.), *Affective Dialog Systems: Proceedings of the ADS 2004 Tutorial and Research Workshop*, Vol. Lecture No, pp. 254–264. Springer, Berlin and Heidelberg.
- Jordan, P. W., & Walker, M. A. (2005). Learning content selection rules for generating object descriptions in dialogue. *Journal of Artificial Intelligence Research*, 24, 157–194.
- Joshi, A. K., & Schabes, Y. (1997). Tree-Adjoining Grammars. In *Handbook of Formal Languages*, Vol. 3, pp. 69–123. Springer, New York.
- Kalchbrenner, N., & Blunsom, P. (2013). Recurrent Continuous Translation Models. In *Proc. EMNLP'13*, pp. 1700–1709.
- Karpathy, A., & Fei-Fei, L. (2015). Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'15)*, pp. 3128–3137.
- Karpathy, A., Joulin, A., & Fei-Fei, L. (2014). Deep Fragment Embeddings for Bidirectional Image Sentence Mapping. In *Proc. NIPS'14*, pp. 1–9.
- Kasper, R. T. (1989). A Flexible Interface for Linking Applications to Penman’s Sentence Generator. In *Proc. Workshop on Speech and Natural Langauge*, pp. 153–158.
- Kauchak, D., & Barzilay, R. (2006). Paraphrasing for automatic evaluation. In *Proc. NAACL-HLT'06*, pp. 455–462.
- Kay, M. (1996). Chart Generation. In *Proc. ACL'96*, pp. 200–204.
- Kazemzadeh, S., Ordonez, V., Matten, M., & Berg, T. (2014). ReferItGame: Referring to Objects in Photographs of Natural Scenes. In *Proc. EMNLP'14*, pp. 787–798.
- Kelleher, J., Costello, F., & Van Genabith, J. (2005). Dynamically structuring, updating and interrelating representations of visual and linguistic discourse context. *Artificial Intelligence*, 167, 62–102.
- Kelleher, J., & Kruijff, G.-J. (2006). Incremental generation of spatial referring expressions in situated dialog. In *Proc. COLING-ACL'06*, pp. 1041–1048.
- Kempen, G. (2009). Clausal coordination and coordinate ellipsis in a model of the speaker. *Linguistics*, 47(3), 653–696.
- Kennedy, C., & McNally, L. (2005). Scale Structure, Degree Modification, and the Semantics of Gradable Predicates. *Language*, 81(2), 345–381.
- Kibble, R., & Power, R. (2004). Optimizing referential coherence in text generation. *Computational Linguistics*, 30(4), 401–416.
- Kiddon, C., & Brun, Y. (2011). That’s what she said: double entendre identification. In *Proc. ACL-HLT'11*, pp. 89–94.

- Kim, J., & Mooney, R. J. (2010). Generative Alignment and Semantic Parsing for Learning from Ambiguous Supervision. In *Proc. COLING'10*, pp. 543–551.
- Kiros, R., Zemel, R. S., & Salakhutdinov, R. (2014). Multimodal Neural Language Models. In *Proc. ICML'14*, pp. 1–14.
- Kojima, A., Tamura, T., & Fukunaga, K. (2002). Natural language description of human activities from video images based on concept hierarchy of actions. *International Journal of Computer Vision*, 50(2), 171–184.
- Koller, A., & Petrick, R. P. (2011). Experiences with planning for natural language generation. *Computational Intelligence*, 27(1), 23–40.
- Koller, A., & Stone, M. (2007). Sentence generation as a planning problem. In *Proc. ACL'07*, pp. 336–343.
- Koller, A., & Striegnitz, K. (2002). Generation as Dependency Parsing. In *Proc. ACL'02*, pp. 17–24.
- Koller, A., Striegnitz, K., Gargett, A., Byron, D., Cassell, J., Dale, R., Moore, J. D., & Oberlander, J. (2010). Report on the second nlg challenge on generating instructions in virtual environments (give-2). In *Proc. INLG'10*, pp. 243–250.
- Koncel-Kedziorski, R., & Hajishirzi, H. (2014). Multi-Resolution Language Grounding with Weak Supervision. In *Proc. EMNLP'14*, pp. 386–396.
- Koncel-Kedziorski, R., Hajishirzi, H., & Farhadi, A. (2014). Multi-resolution language grounding with weak supervision. In *Proc. EMNLP'14*, p. 386396.
- Kondadadi, R., Howald, B., & Schilder, F. (2013). A Statistical NLG Framework for Aggregated Planning and Realization. In *Proc. Volume 1: Long Papers*, pp. 1406–1415.
- Konstas, I., & Lapata, M. (2012). Unsupervised concept-to-text generation with hypergraphs. In *Proc. NAACL-HLT'12*, pp. 752–761.
- Konstas, I., & Lapata, M. (2013). A global model for concept-to-text generation. *Journal of Artificial Intelligence Research*, 48, 305–346.
- Krahmer, E., & Theune, M. (2010). *Empirical Methods in Natural Language Generation*. Springer, Berlin & Heidelberg.
- Krahmer, E., & van Deemter, K. (2012). Computational generation of referring expressions: A survey. *Computational Linguistics*, 38(1), 173–218.
- Krizhevsky, A., Sutskever, I., & Hinton, G. (2012). ImageNet Classification with Deep Convolutional Neural Networks. In Pereira, F., Burges, C. J. C., Bottou, L., & Weinberger, K. Q. (Eds.), *Advances in Neural Information Processing Systems 25*, pp. 1–9. Curran Associates, Inc., USA.
- Kukich, K. (1987). Where do phrases come from: Some preliminary experiments in connectionist phrase generation. In *Natural Language Generation: New Results in Artificial Intelligence, Psychology and Linguistics*. Springer, Berlin and Heidelberg.

- Kukich, K. (1992). Techniques for automatically correcting words in text. *ACM Computing Surveys (CSUR)*, 24(4), 377–439.
- Kulkarni, G., Premraj, V., Dhar, S., Li, S., Choi, Y., Berg, A. C., & Berg, T. (2011). Baby Talk : Understanding and Generating Image Descriptions. In *Proc. CVPR'11*, pp. 1601–1608.
- Kulkarni, G., Premraj, V., Ordonez, V., Dhar, S., Li, S., Choi, Y., Berg, A. C., & Berg, T. (2013). Baby talk: Understanding and generating simple image descriptions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(12), 2891–2903.
- Kutlak, R., Mellish, C., & van Deemter, K. (2013). Content Selection Challenge - University of Aberdeen Entry. In *Proc. ENLG'13*, pp. 208–209.
- Kuznetsova, P., Ordonez, V., Berg, A. C., Berg, T., & Choi, Y. (2012). Collective Generation of Natural Image Descriptions. In *Proc. ACL'12*, pp. 359–368.
- Kuznetsova, P., Ordonez, V., Berg, T., & Choi, Y. (2014). TREETALK: Composition and Compression of Trees for Image Descriptions. *Transactions of the Association for Computational Linguistics*, 2, 351–362.
- Labbé, C., & Portet, F. (2012). Towards an abstractive opinion summarisation of multiple reviews in the tourism domain. In *Proc. International Workshop on Sentiment Discovery from Affective Data*, pp. 87–94.
- Labov, W. (2010). Oral narratives of personal experience. In Hogan, P. C. (Ed.), *Cambridge Encyclopedia of the Language Sciences*, pp. 546–548. Cambridge University Press, Cambridge, UK.
- Lakoff, G., & Johnson, M. (1980). *Metaphors we Live By*. Chicago University Press, Chicago, Ill.
- Langkilde-Geary, I. (2000). Forest-based statistical sentence generation. In *Proc. ANLP-NAACL'00*, pp. 170–177.
- Langkilde-Geary, I., & Knight, K. (2002). HALogen Statistical Sentence Generator. In *Proc. ACL'02 (Demos)*, pp. 102–103.
- Lapata, M. (2006). Automatic Evaluation of Information Ordering: Kendall's Tau. *Computational Linguistics*, 32(4), 471–484.
- Lavie, A., & Agarwal, A. (2007). METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. In *Proc. Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pp. 65–72.
- Lavoie, B., & Rambow, O. (1997). A fast and portable realiser for text generation. In *Proc. ANLP'97*, pp. 265–268.
- Law, A. S., Freer, Y., Hunter, J. R., Logie, R. H., McIntosh, N., & Quinn, J. (2005). A comparison of graphical and textual presentations of time series data to support medical decision making in the neonatal intensive care unit. *Journal of clinical monitoring and computing*, 19(3), 183–94.

- Lebret, R., Grangier, D., & Auli, M. (2016). Generating Text from Structured Data with Application to the Biography Domain. In *Proc. EMNLP'16*.
- LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436–444.
- Lemon, O. (2008). Adaptive Natural Language Generation in Dialogue using Reinforcement Learning. In *Proc. LONDIAL*, p. 141148.
- Lemon, O. (2011). Learning what to say and how to say it: Joint optimisation of spoken dialogue management and natural language generation. *Computer Speech and Language*, 25(2), 210–221.
- Lester, J. C., & Porter, B. W. (1997). Developing and Empirically Evaluating Robust Explanation Generators : The KNIGHT Experiments. *Computational Linguistics*, 23(1), 65–101.
- Levelt, W. (1989). *Speaking: From Intention to Articulation*. MIT Press, Cambridge, MA.
- Levelt, W. (1999). Producing spoken language: a blueprint of the speaker. In Brown, C., & Hagoort, P. (Eds.), *The Neurocognition of Language*, pp. 83–122. Oxford University Press, Oxford and London.
- Levelt, W., Roelofs, A., & Meyer, A. S. (1999). A theory of lexical access in speech production. *The Behavioral and brain sciences*, 22(1), 1–38; discussion 38–75.
- Levenshtein, V. I. (1966). Binary codes capable of correcting deletions, insertions, and reversals. *Soviet Physics Doklady*, 10(8), 707–710.
- Lewis, D. D., & Catlett, J. (1994). Heterogeneous uncertainty sampling for supervised learning. In *Proc. ICML'94*, pp. 148–156.
- Li, S., Kulkarni, G., Berg, T., Berg, A. C., & Choi, Y. (2011). Composing simple image descriptions using web-scale n-grams. In *Proc. CoNLL'11*, pp. 220–228.
- Liang, P., Jordan, M. I., & Klein, D. (2009). Learning Semantic Correspondences with Less Supervision. In *Proc. ACL-IJCNLP'09*, pp. 91–99.
- Lin, C.-Y., & Hovy, E. H. (2003). Automatic Evaluation of Summaries Using N-gram Co-Occurrence Statistics. In *Proc. HLT-NAACL'03*, pp. 71–78.
- Lin, C.-Y., & Och, F. J. (2004). Automatic Evaluation of Machine Translation Quality Using Longest Common Subsequence and Skip-Bigram Statistics. In *Proc. ACL'04*, pp. 605–612.
- Lin, D., & Kong, C. (2015). Generating Multi-sentence Natural Language Descriptions of Indoor Scenes. In *Proc. BMVC'15*, pp. 1–13.
- Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C. L. (2014). Microsoft COCO: Common objects in context. In *Proc. ECCV'14*, Vol. 8693 LNCS, pp. 740–755. Springer.

- Lipschultz, M., Litman, D. J., Jordan, P. W., & Katz, S. (2011). Predicting Changes in Level of Abstraction in Tutor Responses to Students. In *Proc. FLAIRS'11*.
- Lowe, D. G. (2004). Distinctive image features from scale invariant keypoints. *International Journal of Computer Vision*, 60, 91–11020042.
- Lukin, S., & Walker, M. A. (2013). Really? Well. Apparently Bootstrapping Improves the Performance of Sarcasm and Nastiness Classifiers for Online Dialogue. In *Proc. LSM'13*, pp. 30–40.
- Luong, M.-T., Le, Q. V., Sutskever, I., Vinyals, O., & Kaiser, L. (2016). Multi-Task Sequence to Sequence Learning. In *Proc. ICLR'16*, pp. 1–10.
- Luong, M.-T., Socher, R., & Manning, C. D. (2013). Better Word Representations with Recursive Neural Networks for Morphology. In *Proc. CoNLL'13*, pp. 104–113.
- Lutz, T. (1959). Stochastische texte. *Augenblick*, 4(1), 3–9.
- Macdonald, I., & Siddharthan, A. (2016). Summarising news stories for children. In *Proc. INLG'16*, pp. 1–10, Edinburgh, UK.
- Mahamood, S., & Reiter, E. (2011). Generating Affective Natural Language for Parents of Neonatal Infants. In *Proc. ENLG'11*, pp. 12–21.
- Mairesse, F., Gasic, M., Jurcicek, F., Keizer, S., Thompson, B., Yu, K., & Young, S. (2010). Phrase-based statistical language generation using graphical models and active learning. In *Proc. ACL'10*, pp. 1552–1561.
- Mairesse, F., & Walker, M. A. (2010). Towards personality-based user adaptation: Psychologically informed stylistic language generation. *User Modelling and User-Adapted Interaction*, 20(3), 227–278.
- Mairesse, F., & Walker, M. A. (2011). Controlling User Perceptions of Linguistic Style: Trainable Generation of Personality Traits. *Computational Linguistics*, 37(3), 455–488.
- Mairesse, F., & Young, S. (2014). Stochastic language generation in dialogue using factored language models. *Computational Linguistics*, 4(4), 763–799.
- Malinowski, M., Rohrbach, M., & Fritz, M. (2016). Ask your neurons: A neural-based approach to answering questions about images. In *Proc. ICCV'15*, pp. 1–9.
- Mani, I. (2001). *Automatic Summarization*. John Benjamins Publishing Company, Amsterdam.
- Mani, I. (2010). *The Imagined Moment: Time, Narrative and Computation*. University of Nebraska Press, Lincoln, NE.
- Mani, I. (2013). *Computational Modeling of Narrative*. Morgan and Claypool Publishers, USA.
- Mann, W. C., & Matthiessen, C. M. (1983). Nigel: A systemic grammar for text generation (Technical Report RR-83-105). Tech. rep., ISI, University of Southern California, Marina del Rey, CA.

- Mann, W. C., & Moore, J. A. (1981). Computer generation of multiparagraph text. *American Journal of Computational Linguistics*, 7(1), 17–29.
- Mann, W. C., & Thompson, S. A. (1988). Rhetorical structure theory: Toward a functional theory of text organization. *Text*, 8(3), 243–281.
- Manning, C. D. (2015). Last words: Computational linguistics and deep learning. *Computational Linguistics*, 41, 701–707.
- Manurung, R., Ritchie, G. D., & Thompson, H. (2012). Using genetic algorithms to create meaningful poetic text. *Journal of Experimental & Theoretical Artificial Intelligence*, 24(1), 43–64.
- Mao, J., Huang, J., Toshev, A., Camburu, O., Yuille, A., & Murphy, K. (2016). Generation and Comprehension of Unambiguous Object Descriptions. In *Proc. CVPR'16*.
- Mao, J., Xu, W., Yang, Y., Wang, J., Huang, Z., & Yuille, A. (2015a). Deep Captioning with Multimodal Recurrent Neural Networks (m-RNN). In *Proc. ICLR*.
- Mao, J., Xu, W., Yang, Y., Wang, J., Huang, Z., & Yuille, A. (2015b). Learning like a Child: Fast Novel Visual Concept Learning from Sentence Descriptions of Images. In *Proc. ICCV'15*, pp. 2533–2541.
- Marciniak, T., & Strube, M. (2004). Classification-based generation using TAG. In *Proc. INLG'04*, pp. 100–109. Springer.
- Marciniak, T., & Strube, M. (2005). Beyond the Pipeline : Discrete Optimization in NLP. In *Proc. CoNLL'05*, pp. 136–143.
- Martin, J. H. (1990). *A Computational Model of Metaphor Interpretation*. Academic Press, New York.
- Martin, J. H. (1994). Metabank: A knowledge-base of metaphoric language conventions. *Computational Intelligence*, 10(2), 134–149.
- Martinez, H. P., Yannakakis, G. N., & Hallam, J. (2014). Don't classify ratings of affect; Rank Them!. *IEEE Transactions on Affective Computing*, 5(3), 314–326.
- Mason, R., & Charniak, E. (2014). Domain-Specific Image Captioning. In *Proc. CONLL'14*, pp. 11–20.
- Mason, W., & Suri, S. (2012). Conducting behavioral research on amazons mechanical turk. *Behavior Research Methods*, 44(1), 1–23.
- McCoy, K. F., & Strube, M. (1999). Generating Anaphoric Expressions: Pronoun or Definite Description?. In Cristea, D., Ide, N., & Marcu, D. (Eds.), *The Relation of Discourse/Dialogue Structure and Reference: Proceedings of the Workshop held in conjunction with ACL'99*, pp. 63–71.
- McDermott, D. (2000). The 1998 AI planning systems competition. *AI magazine*, 21(2), 1–33.
- McDonald, D. D. (1993). Issues in the Choice of a Source for Natural Language Generation. *Computational Linguistics*, 19(1), 191–197.

- McDonald, D. D. (2010). Natural language generation. In Indurkhy, N., & Damerau, F. (Eds.), *Handbook of Natural Language Processing* (2 edition), p. 121144. Chapman and Hall/CRC, London.
- McDonald, D. D., & Pustejovsky, J. D. (1985). A computational theory of prose style for natural language generation. In *Proc. EACL'85*, pp. 187–193.
- McIntyre, N., & Lapata, M. (2009). Learning to Tell Tales : A Data-driven Approach to Story Generation. In *Proc. ACL-IJCNLP'09*, pp. 217–225.
- McKeown, K. R. (1985). *Text Generation*. Cambridge University Press, Cambridge, UK.
- McRoy, S. W., Channarukul, S., & Ali, S. S. (2003). An augmented template-based approach to text realization. *Natural Language Engineering*, 9(04), 381–420.
- Meehan, J. R. (1977). TALE-SPIN, An Interactive Program that Writes Stories. In *Proc. IJCAI'77*, pp. 91–98. Morgan Kaufmann.
- Mei, H., Bansal, M., & Walter, M. R. (2016). What to talk about and how? Selective Generation using LSTMs with Coarse-to-Fine Alignment. In *NAACL-HLT'16*, pp. 1–11.
- Meister, J. C. (2003). *Computing Action. A Narratological Approach*. Mouton de Gruyter, Berlin.
- Mellish, C., & Dale, R. (1998). Evaluation in the context of natural language generation. *Computer Speech & Language*, 12(4), 349–373.
- Mellish, C., Scott, D., Cahill, L., Paiva, D. S., Evans, R., & Reape, M. (2006). A Reference Architecture for Natural Language Generation Systems. *Natural Language Engineering*, 12(01), 1–34.
- Meteer, M. W. (1991). Bridging the generation gap between text planning and linguistic realization. *Computational Intelligence*, 7(4), 296–304.
- Meteer, M. W., McDonald, D. D., Anderson, S., Forster, D., Gay, L., Iluettnner, A., & Sibun, P. (1987). Mumble-86: Design and Implementation (Technical Report COINS 87-87). Tech. rep., University of Massachusetts at Amherst, Amherst, MA.
- Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Distributed Representations of Words and Phrases and their Compositionalities. In Burges, C. J. C., Bottou, L., Welling, M., Ghahramani, Z., & Weinberger, K. Q. (Eds.), *Advances in Neural Information Processing Systems 26*, pp. 3111–3119. Curran Associates, Inc.
- Mikolov, T., Karafiat, M., Burget, L., Cernocky, J., & Khudanpur, S. (2010). Recurrent Neural Network based Language Model. In *Proc. Interspeech'10*, pp. 1045–1048.
- Miller, G. A. (1995). WordNet: a lexical database for English. *Communications of the ACM*, 38(11), 39–41.

- Mitchell, M., Dodge, J., Goyal, A., Yamaguchi, K., Stratos, K., Han, X., Mensch, A., Berg, A., Han, X., Berg, T., & Daume III, H. (2012). Midge: Generating Image Descriptions From Computer Vision Detections. In *Proc. EACL'12*, pp. 747–756.
- Mitchell, M., van Deemter, K., & Reiter, E. (2013). Generating Expressions that Refer to Visible Objects. In *Proc. NAACL'13*, pp. 1174–1184.
- Mnih, A., & Hinton, G. (2007). Three new graphical models for statistical language modelling. In *Proc. ICML'07*, pp. 641–648.
- Molina, M., Stent, A., & Parodi, E. (2011). Generating Automated News to Explain the Meaning of Sensor Data. In Gama, J., Bradley, E., & Hollmén, J. (Eds.), *Proc. IDA 2011*, pp. 282–293. Springer, Berlin and Heidelberg.
- Montfort, N. (2007). Ordering events in interactive fiction narratives. In *Proc. AAAI Fall Symposium on Intelligent Narrative Technologies*, pp. 87–94.
- Montfort, N. (2013). *World clock*. Harvard Book Store Press, Cambridge, MA.
- Moore, J. D., & Paris, C. (1993). Planning text for advisory dialogues: Capturing intentional and rhetorical information. *Computational Linguistics*, 19(4), 651–694.
- Moore, J. D., Porayska-Pomsta, K., Zinn, C., & Varges, S. (2004). Generating Tutorial Feedback with Affect. In *Proc. FLAIRS'04*.
- Mrabet, Y., Vougiouklis, P., Kilicoglu, H., Gardent, C., Demner-Fushman, D., Hare, J., & Simperl, E. (2016). Aligning Texts and Knowledge Bases with Semantic Sentence Simplification. In *Proc. WebNLG'16*, pp. 29–36.
- Muscat, A., & Belz, A. (2015). Generating Descriptions of Spatial Relations between Objects in Images. In *Proc. ENLG'15*, pp. 100–104.
- Nakanishi, H., Miyao, Y., & Tsujii, J. (2005). Probabilistic Models for Disambiguation of an HPSG-Based Chart Generator. In *Proc. 9th International Workshop on Parsing Technologies*, pp. 93–102.
- Nakatsu, C., & White, M. (2010). Generating with Discourse Combinatory Categorial Grammar. *Linguistic Issues in Language Technology, September*.
- Nauman, A. D., Stirling, T., & Borthwick, A. (2011). What makes writing good? an essential question for teachers. *The Reading Teacher*, 64(5), 318–328.
- Nemhauser, G. L., & Wolsey, L. A. (1988). *Integer programming and combinatorial optimization*. Wiley, Chichester, UK.
- Nenkova, A., & McKeown, K. R. (2011). Automatic Summarization. *Foundations and Trends in Information Retrieval*, 5(2-3), 103–233.
- Netzer, Y., Gabay, D., Goldberg, Y., & Elhadad, M. (2009). Gaiku : Generating Haiku with Word Associations Norms. In *Proc. Workshop on Computational Approaches to Linguistics Creativity*, pp. 32–39.
- Niederhoffer, K. G., & Pennebaker, J. W. (2002). Linguistic Style Matching in Social Interaction. *Journal of Language and Social Psychology*, 21(4), 337–360.

- Nirenburg, S., Lesser, V., & Nyberg, E. (1989). Controlling a language generation planner. In *Proc. IJCAI'89*, pp. 1524–1530.
- Norrick, N. R. (2005). The dark side of tellability. *Narrative Inquiry*, 15(2), 323–343.
- Novikova, J., & Rieser, V. (2016a). The analogue challenge: Non aligned language generation. In *Proc. INLG'16*, pp. 168–170, Edinburgh, UK.
- Novikova, J., & Rieser, V. (2016b). Crowdsourcing NLG Data: Pictures elicit better data. In *Proc. INLG'16*.
- Oberlander, J. (1998). Do the Right Thing ... but Expect the Unexpected. *Computational Linguistics*, 24(3), 501–507.
- Oberlander, J., & Lascarides, A. (1992). Preventing false temporal implicatures: Interactive defaults for text generation. In *Proc. COLING'92*, pp. 721–727.
- Oberlander, J., & Nowson, S. (2006). Whose thumb is it anyway ? Classifying author personality from weblog text. In *Proc. COLING/ACL'06*, pp. 627–634.
- O'Donnell, M. (2001). ILEX: an architecture for a dynamic hypertext generation system. *Natural Language Engineering*, 7(3), 225–250.
- Oh, A. H., & Rudnicky, A. I. (2002). Stochastic natural language generation for spoken dialog systems. *Computer Speech and Language*, 16(3-4), 387–407.
- Oliva, A., & Torralba, A. (2001). Modeling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision*, 42(3), 145–175.
- Ordonez, V., Deng, J., Choi, Y., Berg, A. C., & Berg, T. (2013). From Large Scale Image Categorization to Entry-Level Categories. In *Proc. ICCV'13*, pp. 2768–2775.
- Ordonez, V., Kulkarni, G., & Berg, T. (2011). Im2text: Describing images using 1 million captioned photographs. In *Proc. NIPS'11*, pp. 1143–1151. Curran Associates Ltd.
- Ordonez, V., Liu, W., Deng, J., Choi, Y., Berg, A. C., & Berg, T. (2016). Learning to name objects. *Communications of the ACM*, 59(3), 108–115.
- Orkin, J., & Roy, D. (2007). The restaurant game: Learning social behavior and language from thousands of players online. *Journal of Game Development*, 3, 39–60.
- Ortiz, L. G. M., Wolff, C., & Lapata, M. (2015). Learning to Interpret and Describe Abstract Scenes. In *Proc. NAACL'15*, pp. 1505–1515.
- Paiva, D. S., & Evans, R. (2005). Empirically-based control of natural language generation. In *Proc. ACL'05*, pp. 58–65.
- Pang, B., & Lee, L. (2008). Opinion Mining and Sentiment Analysis. *Foundations and Trends in Information Retrieval*, 1(2), 1–135.

- Papineni, K., Roukos, S., Ward, T., & Zhu, W.-j. (2002). BLEU : a Method for Automatic Evaluation of Machine Translation. In *Proc. ACL'02*, pp. 311–318.
- Passonneau, R. J. (2006). Measuring Agreement on Set-valued Items (MASI) for Semantic and Pragmatic Annotation. In *Proc. LREC'06*, pp. 831–836.
- Pennington, J., Socher, R., & Manning, C. D. (2014). GloVe: Global Vectors for Word Representation. In *Proc. EMNLP'14*.
- Pérez, R., Ortiz, O., Luna, W., Negrete, S., Castellanos, V., Peñalosa, E., & Ávila, R. (2011). A System for Evaluating Novelty in Computer Generated Narratives. In *Proc. ICCC'11*, pp. 63–68.
- Petrovic, S., & Matthews, D. (2013). Unsupervised joke generation from big data. In *Proc. ACL'13*, pp. 228–232.
- Pickering, M. J., & Garrod, S. (2004). Toward a mechanistic psychology of dialogue. *The Behavioral and brain sciences*, 27(2), 169–90; discussion 190–226.
- Pickering, M. J., & Garrod, S. (2013). An integrated theory of language production and comprehension. *The Behavioral and brain sciences*, 36(4), 329–47.
- Piwek, P. (2003). An annotated bibliography of affective natural language generation. Tech. rep., ITRI, University of Brighton.
- Piwek, P., & Boyer, K. E. (2012). Varieties of question generation: Introduction to this special issue. *Dialogue and Discourse*, 3(2), 1–9.
- Plachouras, V., Smiley, C., Bretz, H., Taylor, O., Leidner, J. L., Song, D., & Schilder, F. (2016). Interacting with financial data using natural language. In *Proc. SIGIR'16*, pp. 1121–1124.
- Poesio, M., Stevenson, R., Di Eugenio, B., & Hitzeman, J. (2004). Centering: A parametric theory and its instantiations. *Computational Linguistics*, 30(3), 309–363.
- Portet, F., Reiter, E., Gatt, A., Hunter, J. R., Sripada, S., Freer, Y., & Sykes, C. (2009). Automatic generation of textual summaries from neonatal intensive care data. *Artificial Intelligence*, 173(7-8), 789–816.
- Power, R., Scott, D., & Bouayad-Agha, N. (2003). Document Structure. *Computational Linguistics*, 29(2), 211–260.
- Power, R., & Williams, S. (2012). Generating numerical approximations. *Computational Linguistics*, 38(1), 113–134.
- Propp, V. (1968). *Morphology of the Folk Tale*. University of Texas Press, Austin, TX.
- Rajkumar, R., & White, M. (2011). Linguistically Motivated Complementizer Choice in Surface Realization. In *Proc. UCNLG+Eval'11*, pp. 39–44.
- Rajkumar, R., & White, M. (2014). Better Surface Realization through Psycholinguistics. *Language and Linguistics Compass*, 8(10), 428–448.

- Ramos-Soto, A., Bugarin, A. J., Barro, S., & Taboada, J. (2015). Linguistic Descriptions for Automatic Generation of Textual Short-Term Weather Forecasts on Real Prediction Data. *IEEE Transactions on Fuzzy Systems*, 23(1), 44–57.
- Ratnaparkhi, A. (1996). A maximum entropy model for part-of-speech tagging. In *Proc. EMNLP'96*, pp. 133–142.
- Ratnaparkhi, A. (2000). Trainable methods for surface natural language generation. In *Proc. NAACL'00*, pp. 194–201.
- Reape, M., & Mellish, C. (1999). Just what is aggregation anyway?. In *Proc. ENLG'99*.
- Regneri, M., Rohrbach, M., Wetzel, D., & Thater, S. (2013). Grounding Action Descriptions in Videos. *Transactions of the Association for Computational Linguistics*, 1, 25–36.
- Reiter, E. (1994). Has a consensus NL generation architecture appeared, and is it psycholinguistically plausible?. In *Proc. IWNLG'94*, pp. 163–170.
- Reiter, E. (2000). Pipelines and Size Constraints. *Computational Linguistics*, 26(2), 251–259.
- Reiter, E. (2007). An architecture for data-to-text systems. In *Proc. ENLG'07*, pp. 97–104.
- Reiter, E. (2010). Natural Language Generation. In Clark, A., Fox, C., & Lappin, S. (Eds.), *Handbook of Computational Linguistics and Natural Language Processing*, pp. 574–598. Wiley, Oxford.
- Reiter, E., & Belz, A. (2009). An Investigation into the Validity of Some Metrics for Automatically Evaluating Natural Language Generation Systems. *Computational Linguistics*, 35(4), 529–558.
- Reiter, E., & Dale, R. (1997). Building natural-language generation systems. *Natural Language Engineering*, 3, 57–87.
- Reiter, E., & Dale, R. (2000). *Building Natural Language Generation Systems*. Cambridge University Press, Cambridge, UK.
- Reiter, E., Gatt, A., Portet, F., & van Der Meulen, M. (2008). The Importance of Narrative and Other Lessons from an Evaluation of an NLG System that Summarises Clinical Data. In *Proc. INLG'08*, pp. 147–155.
- Reiter, E., Mellish, C., & Levine, J. (1995). Automatic Generation of Technical Documentation. *Applied Artificial Intelligence*, 9, 259–287.
- Reiter, E., Robertson, R., & Osman, L. M. (2003). Lessons from a failure: Generating tailored smoking cessation letters. *Artificial Intelligence*, 144(1-2), 41–58.
- Reiter, E., & Sripada, S. (2002). Should corpora texts be gold standards for NLG?. In *Proc. INLG'02*, pp. 97–104.

- Reiter, E., Sripada, S., Hunter, J. R., Yu, J., & Davy, I. (2005). Choosing words in computer-generated weather forecasts. *Artificial Intelligence*, 167(1-2), 137–169.
- Riedl, M. O., & Young, R. M. (2005). An objective character believability evaluation procedure for multi-agent story generation systems. In Panayiotopoulos, T., Gratch, J., Aylett, R., Ballin, D., Olivier, P., & Thomas Rist (Eds.), *Proc. 5th International Conference on Intelligent Virtual Agents*.
- Riedl, M. O., & Young, R. M. (2010). Narrative planning: Balancing plot and character. *Journal of Artificial Intelligence Research*, 39, 217–268.
- Rieser, V., Keizer, S., Liu, X., & Lemon, O. (2011). Adaptive Information Presentation for Spoken Dialogue Systems : Evaluation with human subjects. In *Proc. ENLG'11*, pp. 102–109.
- Rieser, V., & Lemon, O. (2009). Natural Language Generation as Planning Under Uncertainty for Spoken Dialogue Systems. In *Eacl 2009*, pp. 683–691.
- Rieser, V., & Lemon, O. (2011a). *Reinforcement Learning for Adaptive Dialogue Systems*. Springer, Berlin and Heidelberg.
- Rieser, V., & Lemon, O. (2011b). *Reinforcement learning for adaptive dialogue systems: a data-driven methodology for dialogue management and natural language generation*. Springer, Berlin & Heidelberg.
- Ritchie, G. D. (2009). Can computers create humor?. *AI Magazine*, 30(3), 71–81.
- Ritter, A., Cherry, C., & Dolan, W. B. (2011). Data-driven response generation in social media. In *Proc. EMNLP'11*, pp. 583–593.
- Robin, J. (1993). A Revision-Based Generation Architecture for Reporting Facts in their Historical Context. In Horacek, H., & Zock, M. (Eds.), *New Concepts in Natural Language Generation: Planning, Realization and Systems*, pp. 238–268. Pinter, London.
- Rowe, J. P., McQuiggan, S. W., Robison, J. L., Marcey, D. R., & Lester, J. C. (2009). STORYEVAL: An Empirical Evaluation Framework for Narrative Generation. In *AAAI Spring Symposium: Intelligent Narrative Technologies II*, pp. 103–110.
- Roy, D. (2002). Learning visually grounded words and syntax for a scene description task. *Computer Speech and Language*, 16(3-4), 353–385.
- Roy, D., & Reiter, E. (2005). Connecting language to the world. *Artificial Intelligence*, 167(1-2), 1–12.
- Rus, V., Piwek, P., Stoyanchev, S., Wyse, B., Lintean, M., & Moldovan, C. (2011). Question generation shared task and evaluation challenge: status report. In *Proc. ENLG'11*, pp. 318–320.

- Rus, V., Wyse, B., Piwek, P., Lintean, M., Stoyanchev, S., & Moldovan, C. (2010). Overview of the first question generation shared task evaluation challenge. In *Proc. 3rd Workshop on Question Generation*, pp. 45–57.
- Schwartz, H. A., Eichstaedt, J. C., Kern, M. L., Dziurzynski, L., Ramones, S. M., Agrawal, M., Shah, A., Kosinski, M., Stillwell, D., Seligman, M. E. P., & Ungar, L. H. (2013). Personality, gender, and age in the language of social media: the open-vocabulary approach. *PloS one*, 8(9), 1–16.
- Schwenk, H., & Gauvain, J.-l. (2005). Training Neural Network Language Models. In *Proc. EMNLP/HLT'05*, pp. 201–208.
- Scott, D., & Sieckenius de Souza, C. (1990). Getting the message across in RST-based text generation. In Dale, R., Mellish, C., & Zock, M. (Eds.), *Current research in natural language generation*, pp. 47–73. Academic Press Professional, Inc., San Diego, CA.
- Searle, J. R. (1969). *Speech Acts: An Essay in the Philosophy of Language*. Cambridge University Press, Cambridge, UK.
- Serban, I. V., Sordoni, A., Bengio, Y., Courville, A., & Pineau, J. (2016). Building End-To-End Dialogue Systems Using Generative Hierarchical Neural Network Models. In *Proc. AAAI*.
- Shaw, J. (1998). Clause aggregation using linguistic knowledge. In *Proc. IWNLG'98*, pp. 138–148.
- Shutova, E., Teufel, S., & Korhonen, A. (2012). Statistical Metaphor Processing. *Computational Linguistics*, 2(2013), 301–353.
- Siddharthan, A. (2014). A survey of research on text simplification. *International Journal of Applied Linguistics*, 165(2), 259–298.
- Siddharthan, A., Green, M., van Deemter, K., Mellish, C., & van der Wal, R. (2013). Blogging birds: Generating narratives about reintroduced species to promote public engagement. In *Proc. INLG'13*, pp. 120–124.
- Siddharthan, A., & Katsos, N. (2012). Offline sentence processing measures for testing readability with users. In *Proc. PITR'12*, pp. 17–24.
- Siddharthan, A., Nenkova, A., & McKeown, K. R. (2011). Information Status Distinctions and Referring Expressions: An Empirical Study of References to People in News Summaries. *Computational Linguistics*, 37(4), 811–842.
- Simonyan, K., & Zisserman, A. (2015). Very Deep Convolutional Networks for Large-Scale Image Recognition. In *Proc. ICLR'15*, pp. 1–10.
- Sleimi, A., & Gardent, C. (2016). Generating Paraphrases from DBpedia using Deep Learning. In *Proc. WebNLG'16*, pp. 54–57.
- Snover, M., Dorr, B., Schwartz, R., Micciulla, L., & Makhoul, J. (2006). A Study of Translation Edit Rate with Targeted Human Annotation. In *Proc. AMTA '06*, pp. 223–231.
- Socher, R., Karpathy, A., Le, Q. V., Manning, C. D., & Ng, A. Y. (2014). Grounded Compositional Semantics for Finding and Describing Images

- with Sentences. *Transactions of the Association for Computational Linguistics (TACL)*, 2(April), 207–218.
- Sparck Jones, K., & Galliers, J. R. (1996). *Evaluating Natural Language Processing Systems: An Analysis and Review*. Springer, Berlin and Heidelberg.
- Sripada, S., Reiter, E., & Davy, I. (2003). SUMTIME-MOUSAM : Configurable Marine Weather Forecast Generator. *Expert Update*, 6(1), 4–10.
- Sripada, S., Reiter, E., & Hawizy, L. (2005). Evaluation of an NLG System using Post-Edit Data: Lessons Learned. In *Proc. ENLG'05*, pp. 133–139.
- Stede, M. (2000). The hyperonym problem revisited: Conceptual and lexical hierarchies in language. In *Proc. INLG'00*, pp. 93–99.
- Steedman, M. (2000). *The Syntactic Process*. MIT Press, Cambridge, MA.
- Steedman, M., & Petrick, R. P. (2007). Planning dialog actions. In *Proc. SIGDIAL'07*, pp. 265–272.
- Stent, A., Marge, M., & Singhai, M. (2005). Evaluating evaluation methods for generation in the presence of variation. In Gelbukh, A. (Ed.), *Proc. CiCLing'05*, Vol. 3406 of *Lecture Notes in Computer Science*, pp. 341–351. Springer Berlin Heidelberg.
- Stent, A., & Molina, M. (2009). Evaluating automatic extraction of rules for sentence plan construction. In *Proc. SIGDIAL'09*, pp. 290–297.
- Stock, O., & Strapparava, C. (2005). The act of creating humorous acronyms. *Applied Artificial Intelligence*, 19(2), 137–151.
- Stock, O., Zancanaro, M., Busetta, P., Callaway, C., Krüger, A., Kruppa, M., Kuflik, T., Not, E., & Rocchi, C. (2007). Adaptive, intelligent presentation of information for the museum visitor in PEACH. *User Modeling and User-Adapted Interaction*, 17(3), 257–304.
- Stoia, L., & Shockley, D. (2006). Noun phrase generation for situated dialogs. In *Proc. INLG'06*, pp. 81–88.
- Stone, M. (2000). On Identifying Sets. In *Proc. INLG'00*, pp. 116–123.
- Stone, M., & Webber, B. (1998). Textual Economy through Close Coupling of Syntax and Semantics. In *Proc. INLG'98*, p. 10.
- Striegnitz, K., Gargett, A., Garoufi, K., Koller, A., & Theune, M. (2011). Report on the second NLG challenge on generating instructions in virtual environments (GIVE-2). In *Proc. ENLG'11*, pp. 243–250.
- Strong, C. R., Mehta, M., Mishra, K., Jones, A., & Ram, A. (2007). Emotionally driven natural language generation for personality rich characters in interactive games. In *Proc. AIIDE'07*, pp. 98–100.
- Sutskever, I., Martens, J., & Hinton, G. (2011). Generating Text with Recurrent Neural Networks. In *Proceedings of the 28th International Conference on Machine Learning (ICML'11)*, pp. 1017–1024.

- Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems 27 (NIPS'14)*, pp. 3104–3112.
- Theune, M., Hielkema, F., & Hendriks, P. (2006). Performing aggregation and ellipsis using discourse structures. *Research on Language and Computation*, 4, 353–375.
- 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.
- Theune, M. (2003). Natural language generation for dialogue: System survey. Tech. rep., Twente University.
- Thomason, J., Venugopalan, S., Guadarrama, S., Saenko, K., & Mooney, R. J. (2014). Integrating Language and Vision to Generate Natural Language Descriptions of Videos in the Wild. In *Proc. COLING'14*, pp. 1218–1227.
- Thompson, H. (1977). Strategy and Tactics: a Model for Language Production. In *Papers from the 13th Regional Meeting of the Chicago Linguistic Society*, Vol. 13, pp. 651–668.
- Tintarev, N., Reiter, E., Black, R., Waller, A., & Reddington, J. (2016). Personal storytelling: Using Natural Language Generation for children with complex communication needs, in the wild. *International Journal of Human Computer Studies*, 92-93, 1–16.
- Togelius, J., Yannakakis, G. N., Stanley, K. O., & Browne, C. (2011). Search-based procedural content generation: A taxonomy and survey. *IEEE Transactions on Computational Intelligence and AI in Games*, 3(3), 172–186.
- Turian, J., Shen, L., & Melamed, I. D. (2003). Evaluation of Machine Translation and its Evaluation. In *Proc. MT Summit IX*, pp. 386–393.
- Turner, R., Sripada, S., Reiter, E., & Davy, I. (2008). Selecting the Content of Textual Descriptions of Geographically Located Events in Spatio-Temporal Weather Data. In *Applications and Innovations in Intelligent Systems XV*, pp. 75–88.
- Turner, S. R. (1992). *MINSTREL: A computer model of creativity and storytelling*. Ph.d. thesis, University of California at Los Angeles.
- van Dalen, A. (2012). The algorithms behind the headlines. *Journalism Practice*, 6(5-6), 648–658.
- van Deemter, K. (2012). *Not exactly: In praise of vagueness*. Oxford University Press, Oxford.
- van Deemter, K. (2016). Designing algorithms for referring with proper names. In *Proc. INLG 2016*, pp. 31–35.
- van Deemter, K., Gatt, A., van der Sluis, I., & Power, R. (2012a). Generation of Referring Expressions: Assessing the Incremental Algorithm. *Cognitive Science*, 36(5), 799–836.

- van Deemter, K., Gatt, A., van Gompel, R. P. G., & Krahmer, E. (2012b). Toward a computational psycholinguistics of reference production. *Topics in cognitive science*, 4(2), 166–83.
- van Deemter, K., Krahmer, E., & Theune, M. (2005). Real versus template-based natural language generation: A false opposition?. *Computational Linguistics*, 31(1), 15–24.
- van Deemter, K., Krenn, B., Piwek, P., Klesen, M., Schröder, M., & Baumann, S. (2008). Fully generated scripted dialogue for embodied agents. *Artificial Intelligence*, 172(10), 1219–1244.
- van der Sluis, I., & Mellish, C. (2010). Towards Empirical Evaluation of Affective Tactical NLG. In Krahmer, E., & Theune, M. (Eds.), *Empirical methods in natural language generation*, pp. 242–263. Springer, Berlin and Heidelberg.
- Varges, S., & Mellish, C. (2010). Instance-based natural language generation. *Natural Language Engineering*, 16(03), 309–346.
- Vaudry, P.-L., & Lapalme, G. (2013). Adapting SimpleNLG for bilingual French-English realisation. In *Proc. ENLG'13*, pp. 183–187.
- Veale, T. (2013). Once More, With Feeling! Using Creative Affective Metaphors to Express Information Needs. In *Proc. ICCM'13*, pp. 16–23.
- Veale, T., & Hao, Y. (2007). Comprehending and Generating Apt Metaphors: A Web-driven, Case-based Approach to Figurative Language. In *Proc. AAAI'07*, pp. 1471–1476.
- Veale, T., & Hao, Y. (2008). A fluid knowledge representation for understanding and generating creative metaphors. In *Proc. COLING'08*, pp. 945–952.
- Veale, T., & Li, G. (2015). Distributed divergent creativity: Computational creative agents at web scale. *Cognitive Computation*, 8(2), 175–186.
- Vedantam, R., Zitnick, C. L., & Parikh, D. (2015). CIDEr: Consensus-based image description evaluation. In *Proc. CVPR'15*, pp. 4566–4575.
- Venigalla, H., & Di Eugenio, B. (2013). UIC-CSC: The Content Selection Challenge Entry from the University of Illinois at Chicago. In *Proc. ENLG'13*, pp. 210–211.
- Venugopalan, S., Rohrbach, M., Darrell, T., Donahue, J., Saenko, K., & Mooney, R. J. (2015a). Sequence to Sequence Video to Text. In *Proc. ICCV'15*, pp. 4534–4542. The Computer Vision Foundation.
- Venugopalan, S., Xu, H., Donahue, J., Rohrbach, M., Mooney, R. J., & Saenko, K. (2015b). Translating Videos to Natural Language Using Deep Recurrent Neural Networks. In *Proc. NAACL'15*, pp. 1494–1504.
- Viethen, J., & Dale, R. (2007). Evaluation in natural language generation: Lessons from referring expression generation. *Traitement Automatique des Langues*, 48(1), 141–160.
- Viethen, J., & Dale, R. (2008). The Use of Spatial Relations in Referring Expression Generation. In *Proc. INLG'08*, pp. 59–67.

- Viethen, J., & Dale, R. (2010). Speaker-dependent variation in content selection for referring expression generation. In *Proc. 8th Australasian Language Technology Workshop*, pp. 81–89.
- Viethen, J., & Dale, R. (2011). GRE3D7: A Corpus of Distinguishing Descriptions for Objects in Visual Scenes. In *Proc. UCNLG+Eval'11*, pp. 12–22.
- Vinyals, O., Toshev, A., Bengio, S., & Erhan, D. (2015). Show and Tell: A Neural Image Caption Generator. *CoRR*, *abs/1411.4*.
- Wah, C., Branson, S., Welinder, P., Perona, P., & Belongie, S. (2011). The Caltech-UCSD Birds-200-2011 Dataset (Technical Report CNS-TR-2011-001). Tech. rep., California Institute of Technology, California.
- Walker, M. A. (1992). Redundancy in Collaborative Dialogue. In *Proc. COLING'92*, pp. 345–351.
- Walker, M. A., Cahn, J. E., & Whittaker, S. J. (1997). Improvising linguistic style: Social and affective bases for agent personality. In *Proc. Agents'97*, pp. 96–105.
- Walker, M. A., Park, F., Rambow, O., & Rogati, M. (2001). SPoT : A Trainable Sentence Planner. In *Proc. NAACL'01*, pp. 1–8.
- Walker, M. A., Rambow, O., & Rogati, M. (2002). Training a sentence planner for spoken dialogue using boosting. *Computer Speech and Language*, *16*(3–4), 409–433.
- Walker, M. A., Stent, A., Mairesse, F., & Prasad, R. (2007a). Individual and domain adaptation in sentence planning for dialogue. *Journal of Artificial Intelligence Research*, *30*, 413–456.
- Walker, M. A., Stent, A., Mairesse, F., & Prasad, R. (2007b). Individual and domain adaptation in sentence planning for dialogue. *Journal of Artificial Intelligence Research (JAIR)*, *30*, 413–456.
- Waller, A., Black, R., OMara, D. a., Pain, H., Ritchie, G. D., & Manurung, R. (2009). Evaluating the STANDUP Pun Generating Software with Children with Cerebral Palsy. *ACM Transactions on Accessible Computing*, *1*(3), 1–27.
- Wang, J., & Gaizauskas, R. (2015). Generating Image Descriptions with Gold Standard Visual Inputs : Motivation , Evaluation and Baselines. In *Proc. ENLG'15*, pp. 117–126.
- Wang, L., Raghavan, H., Cardie, C., & Castelli, V. (2014). Query-Focused Opinion Summarization for User-Generated Content. In *Proc. COLING '14*, pp. 1660–1669.
- Wanner, L. (2010). Report generation. In Indurkhya, N., & Damerau, F. (Eds.), *Handbook of Natural Language Processing* (2 edition), pp. 533–555. Chapman and Hall/CRC, London.
- Wanner, L., Bosch, H., Bouayad-Agha, N., & Casamayor, G. (2015). Getting the environmental information across: from the Web to the user. *Expert Systems*, *32*(3), 405–432.

- Wen, T.-h., Gasic, M., Mrksić, N., Su, P.-h., Vandyke, D., & Young, S. (2015). Semantically Conditioned LSTM-based Natural Language Generation for Spoken Dialogue Systems. In *Proc. EMNLP'15*, pp. 1711–1721.
- White, M., Clark, R. A. J., & Moore, J. D. (2010). Generating tailored, comparative descriptions with contextually appropriate intonation. *Computational Linguistics*, 36(2), 159–201.
- White, M., & Howcroft, D. M. (2015). Inducing Clause-Combining Rules : A Case Study with the SPaRKy Restaurant Corpus. In *Proc. ENLG'15*, pp. 28–37.
- White, M., & Rajkumar, R. (2009). Perceptron reranking for CCG realization. In *Proc. EMNLP'09*, pp. 410–419, Singapore.
- White, M., & Rajkumar, R. (2012). Minimal dependency length in realization ranking. In *Proc. EMNLP'12*, pp. 244–255, Jeju Island, Korea.
- White, M., Rajkumar, R., & Martin, S. (2007). Towards Broad Coverage Surface Realization with CCG. In *Proc. UCNLG+MT*.
- Wilks, Y. (1978). Making preferences more active. *Artificial Intelligence*, 11(3), 197–223.
- Williams, S., & Reiter, E. (2008). Generating basic skills reports for low-skilled readers. *Natural Language Engineering*, 14(4), 495–525.
- Winograd, T. (1972). Understanding natural language. *Cognitive Psychology*, 3(1), 1–191.
- Wong, M. T., Hon, A., & Chun, W. (2008). Automatic Haiku Generation Using VSM. In *Proc. ACACOS'08*, pp. 318–323.
- Wong, Y. W., & Mooney, R. J. (2007). Generation by Inverting a Semantic Parser That Uses Statistical Machine Translation. In *Proc. NAACL-HLT'07*, pp. 172–179.
- Wubben, S., van den Bosch, A., & Krahmer, E. (2012). Sentence Simplification by Monolingual Machine Translation. In *Proc. ACL'12*, pp. 1015–1024.
- Xu, K., Ba, J. L., Kiros, R., Cho, K., Courville, A., Salakhutdinov, R., Zemel, R. S., & Bengio, Y. (2015). Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. In *Proc. ICLR'15*.
- Yagcioglu, S., Erdem, E., & Erdem, A. (2015). A Distributed Representation Based Query Expansion Approach for Image Captioning. In *Proc. ACL-IJCNLP'15*, pp. 106–111.
- Yang, Y., Teo, C. L., Daume III, H., & Aloimonos, Y. (2011). Corpus-Guided Sentence Generation of Natural Images. In *Proc. EMNLP'11*, pp. 444–454.
- Yannakakis, G. N., & Martínez, H. P. (2015). Ratings are Overrated!. *Frontiers in ICT*, 2(July).
- Yao, B. Z., Yang, X., Lin, L., Lee, M. W., & Zhu, S. C. (2010). I2T: Image parsing to text description. *Proceedings of the IEEE*, 98(8), 1485–1508.

- Yatskar, M., Galley, M., Vanderwende, L., & Zettlemoyer, L. (2014). See No Evil , Say No Evil : Description Generation from Densely Labeled Images. In *Proc. *SEM*.
- Young, P., Lai, A., Hodosh, M., & Hockenmaier, J. (2014). From Image Descriptions to Visual Denotations: New Similarity Metrics for Semantic Inference over Event Descriptions. *Transactions of the Association for Computational Linguistics (TACL)*, 2(April), 67–78.
- Young, R. M. (2008). Computational Creativity in Narrative Generation: Utility and Novelty Based on Models of Story Comprehension. In *Creative Intelligent Systems, Papers from the 2008 AAAI Spring Symposium (Technical Report SS-08-03)*, pp. 149–155.
- Youyou, W., Kosinski, M., & Stillwell, D. (2015). Computer-based personality judgments are more accurate than those made by humans. *Proceedings of the National Academy of Sciences*, 112(4), 1036–1040.
- Yu, C., & Ballard, D. H. (2004). A multimodal learning interface for grounding spoken language in sensory perceptions. *ACM Transactions on Applied Perception (TAP)*, 1(1), 57–80.
- Yu, H., & Siskind, J. M. (2013). Grounded language learning from video described with sentences. In *Proc. ACL'13*, pp. 53–63.
- Yu, J., Reiter, E., Hunter, J. R., & Mellish, C. (2006). Choosing the content of textual summaries of large time-series data sets. *Natural Language Engineering*, 13(01), 25.
- Zaremba, W., Sutskever, I., & Vinyals, O. (2015). Recurrent Neural Network Regularization. *arXiv preprint, arXiv:1409*.
- Zarrieß, S., & Kuhn, J. (2013). Combining Referring Expression Generation and Surface Realization: A Corpus-Based Investigation of Architectures. In *Proc. ACL'13*, pp. 1547–1557.
- Zarrieß, S., Loth, S., & Schlangen, D. (2015). Reading Times Predict the Quality of Generated Text Above and Beyond Human Ratings. In *Proc. ENLG'15*, pp. 38–47.
- Zhang, X., & Lapata, M. (2014a). Chinese Poetry Generation with Recurrent Neural Networks. In *Proc. EMNLP'14*, pp. 670–680.
- Zhang, X., & Lapata, M. (2014b). Chinese poetry generation with recurrent neural networks. In *EMNLP*, pp. 670–680.
- Zhu, J. (2012). Towards a Mixed Evaluation Approach for Computational Narrative Systems. In *Proc. ICCC'12*, pp. 150–154.
- Zitnick, C. L., & Parikh, D. (2013). Bringing semantics into focus using visual abstraction. In *Proc. CVPR'13*, pp. 3009–3016.
- Zitnick, C. L., Parikh, D., & Vanderwende, L. (2013). Learning the Visual Interpretation of Sentences. In *Proc. ICCV'13*, pp. 1681–1688.