Sentence Generation from Conceptual Graphs

Nicolas Nicolov, Chris Mellish, Graeme Ritchie

Dept of Artificial Intelligence
University of Edinburgh
80 South Bridge
Edinburgh EH1 1HN
{nicholas,chrism,graeme}@aisb.ed.ac.uk

Abstract. This paper describes a technique for translating the semantic information encoded in a conceptual graph into an English language sentence. The use of a non-hierarchically structured semantic representation (conceptual graphs) allows us to investigate a more general version of the sentence generation problem where one is not pre-committed to a choice of the syntactically prominent elements in the initial semantics. We show clearly how the semantic structure is declaratively related to linguistically motivated syntactic representation. Our technique provides flexibility to address cases where the entire input cannot be precisely expressed in a single sentence.

1 Introduction

Natural language generation is the process of realising communicative intentions as text (or speech). The generation task is standardly broken down into the following processes: content determination (what is the meaning to be conveyed), sentence planning (chunking the meaning into sentence sized units, choosing words), surface realisation (determining the syntactic structure), morphology (inflection of words), synthesising speech or formatting the text output.

In this paper we address aspects of sentence planning (how content words are chosen but not how the semantics is chunked in units realisable as sentences) and surface realisation (how syntactic structures are computed). We thus discuss what in the literature is sometimes referred to as tactical generation, that is "how to say it"—as opposed to strategic generation—"what to say". We look at ways of realising a non-hierarchical semantic representation (conceptual graph) as a sentence, and explore the interactions between syntax and semantics.

This work improves on existing generation approaches in the following respects: (i) Unlike the majority of generators this one takes a non-hierarchical semantic representation as its input. This allows us to look at a more general version of the realisation problem which in turn has direct ramifications for the increased paraphrasing power of the generator; (ii) The generator can happen to convey more information than is originally specified in its semantic input. We have a principled way to account for such additions; (iii) We can make finer distinctions as to what counts as a valid rendition of the input semantics by building the corresponding semantics of the generated sentence and exploring
how close this structure is to the original input semantics; (iv) We show how the semantics is systematically related to syntactic structures in a declarative framework. Alternative processing strategies using the same knowledge sources can therefore be envisaged.

Before giving a more detailed description of our proposals first we review some approaches to generation from semantic networks (Section 2). We proceed with some background about the grammatical framework we will employ—Tree Adjoining Grammars (Section 3) and after describing the knowledge sources available to the generator (Section 4) we present the generation algorithm (Section 5). This is followed by a step by step illustration of the generation of one sentence (Section 6). We then discuss further semantic aspects of the generation (Section 7) and the implementation (Section 8). We conclude with a discussion of some issues related to the proposed technique (Section 9). The actual generation algorithm is presented in the appendix.

2 Generation from Semantic Networks

The input for generation systems varies radically from system to system. Most generators expect their input to be cast in a tree-like notation which enables the actual systems to assume that nodes higher in the semantic structure are more prominent than lower nodes. The semantic representations used are variations of a predicate with its arguments. The predicate is realised as the main verb of the sentence and the arguments are realised as complements of the main verb—thus the control information is to a large extent encoded in the tree-like semantic structure. Unfortunately, such dominance relationships between nodes in the semantics often stem from language considerations and are not always preserved across languages. Moreover, if the semantic input comes from other applications, it is hard for these applications to determine the most prominent concepts because linguistic knowledge is crucial for this task. The tree-like semantics assumption leads to simplifications which reduce the paraphrasing power of the generator (especially in the context of multilingual generation). In contrast, the use of a non-hierarchical representation for the underlying semantics allows the input to contain as few language commitments as possible and makes it possible to address the generation strategy from an unbiased position. We have chosen conceptual graphs (cGs) to represent the input to our generator. This has the added advantage that the representation has well defined deductive mechanisms.

The use of semantic networks in generation is not new [15, 13]. It is surprising that although cGs were developed to express natural language semantics and in the seminal work [17] section 5.4 is entirely devoted to surface realisation there has been little work that has taken up this line of research.

Two main approaches have been employed for generation from semantic networks: utterance path and incremental consumption. An utterance path is the sequence of nodes and arcs that are traversed in the process of mapping a graph to a sentence. Generation is performed by finding a cyclic path in the graph which visits each node at least once. If a node is visited more than once, grammar