

LANGUAGE SYNTHESIS USING IMAGE PLACEMENT

S.A.M.M. DE KONINK

FNWI, University of Amsterdam

stefan@konink.de

Outlined images can be used for communication between users. Using the combination of semantic knowledge, scenes having user initiated motion, discretised gauges, and placement of these images present an opportunity to machine translate a storyboard into a paragraph. This paragraph can be synthesised into speech. The story results in a target language free representation, without prior knowledge of language, other than the cultural understanding of the scene. In compare to other methods based on pictogram's our approach does not require a priory knowledge of the target language and is language independent.

1. Introduction

Previous approaches on pictogram language synthesis use icons for word selection. A limited set of available and usually theme based icons are presented to the user. Before dynamic screens were available the limitation of the amount of icons, and thus expression, was increased by the use of multiple combinations of icons to represent one word. A commercial implementation of this state machine is MinSpeak (2005). As example; colour selection is an association of a colour modifier and a specific icon. 'Red' would therefore be represented as 'COLOUR' + 'RED APPLE'. While the use of 'I eat an apple' becomes 'I' + 'EAT' + 'APPLE'. The prediction of sentences in the different systems and the use of morphology depends greatly on vendor specific implementations, but all have one thing in common, they are sequential and assume natural language knowledge.

The method we present is based on the idea of sketching a story, in our current model we removed 'sketch recognition' from the scope and present a set of theme based outlines of images instead. This will decrease the code complexity for our prototype interface significantly, while maintaining the strength of showing what would happen if a user had drawn this image. The placement and motion represent the where and what in a scene. While the dynamic view port targets the subject that is in view and its properties. Specific sentences based on the content in the scene can be put on a story line, representing a sub-paragraph. We put no claims on efficiency of writing, we merely show the ability to read, describe and translate concepts outside the scope of written language.

2. Implementation

In order to use an image driven interface for natural language processing we introduce a very basic interface for this operation. The interface is merely a visual representation of an ontology based on positional information of images, related to each other. The rendering of them will be beneficial for the average user, but can be omitted completely for the actual semantics retrieved from them. The visual rendering within this interface we will describe as *concepts*. A concept is typically something materialised; it has a defined shape, a visual representation and sub positional information to address the touching of two concepts on a plane. The concept identifier will be related to ‘world knowledge’ about this concept. The storage within the interface will use an instance of a concept to be within a position with a state. The state can be a vector of motion with respect to the interface. For each concept, properties can be learnt globally to make them available between all sessions of the program, or locally to adapt an active instance of the concept. Concepts are language independent, for each concept a global ontology, such as a multi-lingual WordNet, takes care of the factual generation of spoken and written language. This will without a doubt require the use of annotated imagery when new concepts are defined.

An instance is placed within a *scene*. The scene fills up the interface, and will show the user an overview of several objects that can form a paragraph when described. A user is able to sub select or zoom into a specific part of a scene, a *subscene*, to describe it into more details or ignore the rest of the scene into the present generated sentence. Within this scene instances can be moved according to laws of physics. They can be stacked on top of each other, and in principle use a virtual horizon with and an artificial surface to build from.

In order to describe a story of more than one paragraph about a present state, we introduce the notion of a *timeline*. The timeline will store all described sub-scenes as non-destructive copies. The timeline will sequence paragraphs with a typical ‘comic book’ look and feel. Within the ontology domain the timeline is just a set of references to objects and their state within a certain time span, that will loop for the amount of time a user has annotated in imagery and motion.

2.1. Storage model and Language synthesis

We take off from a standpoint that there will be an interface that is able to generate an XML file of a scene. The current storage model will store every concept, modifier and world knowledge into XML files for processing. In order to infer sentences from image representations with abstract positional and movement information, we will use an ontology based approach where an XSLT transformation is defined. This will allow us to map the incoming instances to known concepts where we try to find as a priory defined relation between concepts. This notion of semantics can deduce, within a statistical framework, what the user might want to

express.

The XSLT function describes a non-sophisticated function to map input into a predefined output template. In this same transformation step the emphasis is added as meta-data for the generation of spoken language. We will follow Reither and Dale (2000) as outlined in Theune (2003) and Hjalmarsson (2006). The actual design, implementation and input data can be found as the appendix of this paper.

2.1.1. *Document Planning*

Content determination is implemented as XML file generation, either by hand or by an application. The XML format describes the objects used for the basic generation task. Using a delta presentation between scenes, objects that loose or gain their presence between scenes can be gathered. A semantic representation between objects and the sequence they are described in, plus the placement of different (sub-)scenes on the timeline will determine the final *document structure*. The implementation will structure discourse using a set of semantic relationships present in the ontology, the application has a mapping of a representation of these relationships in a physical simulated world.

2.1.2. *Microplanning*

An annotated concept has a place in the ontology that describes the relationships between words, and their lexical forms. The *lexicalisation* tasks can use the label in the ontology as result or query external references by means of WordNet (Fellbaum, 1998). The target audience might prefer a richer use of a language or the opposite a smaller use that are already learnt depending on the familiarity with the language the system is working in. Although style could be semantically related, we will currently ignore specific user preferences at lexicalisation. *Aggregation*, the combination and ordering of utterances is realised by the merging of the inputs as it affects the same objects, and thus shares syntax in XML.

2.1.3. *Surface realisation*

To translate concepts into text to facilitate *linguistic realisation* the ontology and verb sources are queried. Using the processed concepts as input the output will result in morphological corrections. The grammatically correctness and *structure realisation* is forced upon the input using the grammar template file.

The structure of the output that is created follows the SSML (2004) standard. This allows the paragraph to be synthesised into speech. For our work we have used eSpeak (2008b) to Mbrola (2008c) output. Where the latter one synthesises to a PCM wave file. Because the gender of the concepts is available it might be possible to synthesise conversations in the appropriate gender.

3. Discussion

It is clear for us that not every possible attribute or action of a concept is available in an easy accessible format, but even if it was, learning of new concepts and their representation would still be a multi-lingual nightmare. A possible solution without using users that speak two or more foreign language, would be a description of this concept by two users that both use a different mother tongue. If these two users are paired up in describing, the concepts attribute or action, visually and explicitly entering the new concept by spoken language or the keyboard, new concepts can be added without predefined semantics to extend the ontology, this will allow the modern age 'Talking hands and feet'.

The lack of actions will probably limit the application in the first state of usage, after this period the guessing can become ambiguous, and must incorporate sentence disambiguation. Because in the scenes a predefined world is present the relation of two objects might be either not defined, present, or overdefined. In the first case the definition should be made, whether in the last case multiple solutions are possible and should be shown or vocalised to the user. The choice a user makes at that point is stored on top of two instances but should increase the likelihood of this relation because of its statistical presence. Once up to speed, language by images, will just behave as any other language now known in the textual domain. It will not be anymore explicit, but more people should be able to read it because it uses an assumption no other language uses: social and cultural knowledge opposed to the recognition of characters, words, lexicon and grammar.

4. Future work

As described we use a non-sophisticated function to map input to an output template. By actually making a relevant choice of possible word sequences based on a tree structure it must be possible to write out a higher quality of language into different languages. The use of proprietary formats in this effort does not help to standardise such an effort. Using de facto standards such as NaturalOWL for storage and processing on top of the intermediate data, more extensible generation is possible. The creation of such formal ontology might not only benefit one user, but shared via the web, allows the co-creation and reuse of new semantics for every client that is connected.

We did not implement a graphical interface, a future research project must be able to implement such interface by using the boundaries set in this paper. The interface can then be plugged before the XML processing and therefore extend this work, rather than to supersede it.

5. Conclusion

In this document we have shown the basic possibilities for the generation of language independent output as in text, as in speech, by using imagery input and relations to each other. While the actual implementation of a graphic interface must show if the usage is practical, this paper might be a starting point for further work.

Acknowledgements

I would like to thank Rob van Son for stimulating me to write this paper and motivate me to improve it.

License

This complete work is licensed under the Creative Commons - Non Commercial, Attribution, Share Alike (CC, 2008a). Other licensing is only permitted with a written consent of the author.

References

- <http://www.w3.org/TR/speech-synthesis/>. (2004).
- <http://www.minspeak.com/>. (2005).
- <http://creativecommons.org/licenses/by-nc-sa/2.0/>. (2008a).
- <http://espeak.sourceforge.net/>. (2008b).
- <http://tcts.fpms.ac.be/synthesis/mbrola.html>. (2008c).
- Fellbaum, C. (Ed.). (1998). *WordNet: an electronic lexical database*. MIT Press.
- Hjalmarsson, A. (2006). Utterance generation in spoken dialog systems.
- Reither, E., & Dale, R. (2000). Building natural language generation systems.
- Theune, M. (2003). Natural language generation for dialogue: system survey.

Appendix A. Design

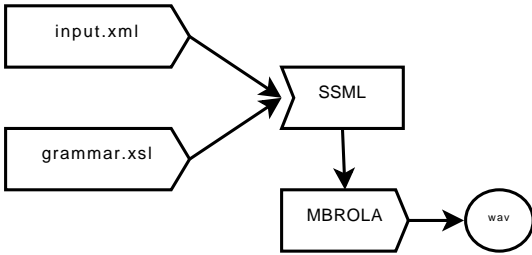


Figure 1. The program flow

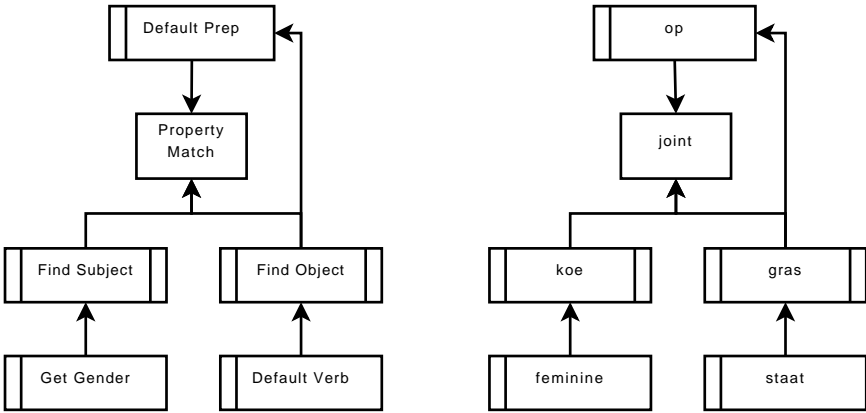


Figure 2. The input transformation

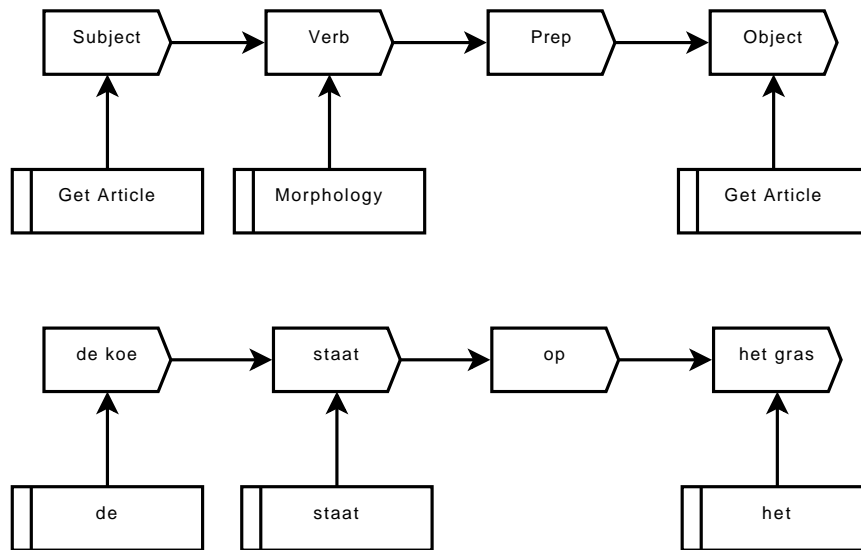


Figure 3. The language generation

Appendix B. XML Input

Appendix B.1. *Generated from the interface*

```
<story>
  <joint>
    <actor title="koe" />
    <actor title="gras" />
  </joint>
</story>
```

Appendix B.2. *Knowledge Ontology*

```
<knowledge>
  <subject title="koe">
    <property prop="gender" value="she" />
    <property food="gras" />
    <property prop="defaultart" value="de" />
  </subject>
  <subject title="gras">
    <property colour="groen" />
    <property prop="defaultprep" value="op" />
    <property prop="defaultart" value="het" />
    <property prop="defaultverb" value="staan" />
  </subject>
</knowledge>
```

Appendix B.3. *Verb Morphology*

```
<verbs>
  <verb name="staan">
    <alt verv="me" value="sta" />
    <alt verv="he" value="staat" />
    <alt verv="it" value="staat" />
    <alt verv="she" value="staat" />
  </verb>
</verbs>
```

Appendix B.4. *Grammar order*

```
<grammar language="dutch">
  <rule>
    <subject />
    <verb />
    <object />
  </rule>
</grammar>
```

Appendix B.5. *Grammar and structure template*

```
<?xml version="1.0"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL
/Transform">
<xsl:output method="html" indent="yes"/>
<xsl:template match="/">
<speak xmlns="http://www.w3.org/2001/10/synthesis" version="1.0">
<break time="3" />
<prosody rate="1.2">
<xsl:apply-templates />
</prosody>
</speak>
</xsl:template>

<xsl:template match="joint">
<xsl:variable name="subject" select="*[1]/@title" />
<xsl:variable name="object" select="*[2]/@title" />
<xsl:variable name="gender" select="document('knowledge.xml')/
knowledge/subject[@title=$subject]/property[@prop='gender']/
@value" />
<xsl:variable name="defaultverb" select="document('knowledge.xml')
/knowledge/subject[@title=$object]/property[@prop='defaultverb
']/@value" />
<s>
<emphasis level="none"><xsl:value-of select="document('knowledge
.xml')/knowledge/subject[@title=$subject]/property[@prop='
defaultart']/@value" /></emphasis>
<emphasis level="moderate"><xsl:value-of select="$subject" /></
emphasis>
<emphasis level="reduced"><xsl:value-of select="document('verbs.
xml')/verbs/verb[@name=$defaultverb]/alt[@verv=$gender]/
```

```
        @value" /></emphasis>
    <emphasis level="reduced"><xsl:value-of select="document('
        knowledge.xml')/knowledge/subject[@title=$object]/property[
        @prop='defaultprep']/@value" /></emphasis>
    <emphasis level="reduced"><xsl:value-of select="document('
        knowledge.xml')/knowledge/subject[@title=$object]/property[
        @prop='defaultart']/@value" /></emphasis>
    <emphasis level="moderate"><xsl:value-of select="$object" /></
    emphasis>
</s>
</xsl:template>

<xsl:template match="story">
<p>
    <xsl:apply-templates />
</p>
</xsl:template>
</xsl:stylesheet>
```