Portuguese Text Generation from Large Corpora

Eder M. de Novais, Ivandré Paraboni, Douglas F. P. da Silva Junior

School of Arts, Sciences and Humanities, University of São Paulo
Av. Arlindo Bettio, 1000 - São Paulo, Brazil
eder.novais@usp.br, ivandre@usp.br, douglas.fernandes.silva@usp.br

Abstract

In the implementation of a surface realisation engine, many of the computational techniques seen in other AI fields have been widely applied. Among these, the use of statistical methods has been particularly successful, as in the so-called 'generate-and-select', or 2-stages architectures. Systems of this kind produce output strings from possibly underspecified input data by over-generating a large number of alternative realisations (often including ungrammatical candidate sentences.) These are subsequently ranked with the aid of a statistical language model, and the most likely candidate is selected as the output string. Statistical approaches may however face a number of difficulties. Among these, there is the issue of data sparseness, a problem that is particularly evident in cases such as our target language - Brazilian Portuguese - which is not only morphologically-rich, but relatively poor in NLP resources such as large, publicly available corpora. In this work we describe a first implementation of a shallow surface realisation system for this language that deals with the issue of data sparseness by making use of factored language models built from a (relatively) large corpus of Brazilian newspapers articles.

Keywords: Text Generation, Surface Realisation, Language Modelling

1. Introduction

In a standard Natural Language Generation (NLG) architecture (Reiter, 2007), the surface realisation task consists of taking abstract sentence specifications as an input and generating output strings in a given target language. Assuming that other tasks such as lexical choice have already been performed, surface realisation proper (i.e., as implemented by a surface realisation engine (Gatt and Reiter, 2009)) focuses on language-dependent tasks such as performing constituents agreement and sentence linearization. Surface realisation is an active research topic in NLG, and it has been recently the focus of the first Surface Realisation Challenge Task competition (Belz et al., 2011)..

In the implementation of a surface realisation engine, many of the computational techniques seen in other AI fields have been widely applied. Among these, the use of statistical methods has been particularly successful, as in the so-called 'generate-and-select', or 2-stages NLG architectures introduced in Langkilde, (2000). Systems of this kind produce output strings from possibly underspecified input data by over-generating a large number of alternative realisations (often including ungrammatical candidate sentences.) These are subsequently ranked with the aid of a statistical language model, and the most likely candidate is selected as the output string.

Statistical approaches to NLG may however face a number of difficulties. Among these, there is the issue of data sparseness, a problem that is particularly evident in cases such as our target language - Brazilian Portuguese - which is not only morphologically-rich, but relatively poor in NLP resources such as large, publicly available corpora. In this work we describe an implementation of a shallow surface realisation system for this language that deals with the issue of data sparseness by making use of factored language models (Bilmes and Kirchhoff, 2003) built from a (relatively) large corpus of Brazilian newspapers articles.

The reminder of this paper is organised as follows. Section 2 presents an overview of our system, and section 3 describes the language models under consideration. Section 4 presents the results of the evaluation work. Section 5 describes related work and section 6 draws conclusions.

2. System Overview

Our system takes as an input an unordered dependency tree representing the sentence to be generated, in which all content words have been previously determined. The input data may optionally convey information such as gender, number, tense, definiteness etc. but in case the underlying application is not able to provide these details, default values will be applied and subsequently adjusted with the aid of dictionary information described in (Muniz et al., 2005).

The following is an example (adapted from Portuguese) of possible input to our system including gender (m), and definiteness (def) information, but not number. This input could be realised as the NP 'the Indian writer Amitav Ghosh'.

> **Input** : concept(s15, [['amitav=ghosh'], ['indian'], head('writer'), def, m]).
>
> **Output** : 'the Indian writer Amitav Ghosh' / 'o escritor indiano Amitav Ghosh'

Example 1 - Input specification and expected output

The input is processed by selecting an appropriate sentence template (e.g., active or passive voice etc.) and then treating each sentence constituent (Agent, Action and Patient) individually, starting with Agent.

For each constituent, head terms are scanned and their features, if available, are enforced to all subordinated terms. For instance, the male (m) gender in the head term of Example 1 guarantees that all terms within concept s15 will have the same gender and, if necessary, the same feature will be applied to other sentence constituents (e.g., verb complement etc. not shown in the example.)

If the information provided by the head term is incomplete, the system will attempt to inherit a non-ambiguous feature from the existing terms with the aid of the dictionary. For instance, the search for non-ambiguos gender/number features of concept NP terms starts by looking for proper names (which are more likely to have a single gender/number value) and, if necessary, considering nouns and modifiers, in that order. Thus, the missing number feature for the NP in Example 1 may be inherited from the information provided by the (singular) proper name 'Amitav Ghosh', if available from the dictionary. On the other hand, if the required information is not available, or if the term under consideration is ambiguos (e.g., as most nouns, 'writer' has distinct male and female forms in Portuguese) the system will attempt to inherit from any other constituent.

Finally, if there is still any missing information after considering all alternatives, the system will select a default value consistent with all terms. If no such value is available (e.g., if some terms can only be expressed in singular form, and others in plural) the input is considered invalid.

Once all relevant features of the Agent constituent have been determined, Action features are determined first by inheriting (likewise in the English language) the verb number feature from the Agent constituent. Next, other features (mode, tense etc.) are determined using the same rules as above (i.e., by inheritance and/or use of default values.)

Patient features are determined in the same way as in the case of the Agent, or, in the particular case of auxiliary verb usage (e.g., 'She is beautiful') Portuguese grammar requires the verb complement to agree with the subject, that is, Patient gender and number information are also inherited from the Agent constituent (making the female form 'Ela é bonita', and not 'Ela é #bonito', which is ungrammatical).

After agreement has been established, all that remains to be done is sentence linearization. In our current implementation, only active voice sentences in the form <NP VP NP> are supported. Thus, given that we have a fixed sentence ordering, the actual linearization to be performed is a matter of finding the correct order of VP and NP constituents.

Writing linearization rules for VPs can be costly, e.g., if a large number of target languages are considered, and the task may become even more complex in the case of the ordering of noun modifiers (Malouf, 2000; Mitchell, 2009). For that reason, instead of handcrafting linearization rules for VPs and NPs in every target language, we follow (Langkilde, 2000) and others and leave the task to be decided by a statistical language model. More specifically, we over-generate all possible permutations of NP and VP constituents and select the most likely output with the aid of a language model.

3. Language Modelling

Statistical NLP for morphologically-rich languages may make more explicit the issue of data sparseness, possibly requiring much larger amounts of data to obtain results that are comparable to those observed in languages such as English (Novais et al., 2011) In the case of our target language - Brazilian Portuguese - the largest corpus that is publicly available for research purposes is the 32 million word NILC corpus in (Nunes et al., 1996), which is clearly insufficient for our purposes.

As a means to overcome these limitations, we collected additional documents from the web and created an extended 142-million words corpus of Brazilian newspapers articles. This is, to our knowledge, the largest corpus of this kind for Brazilian Portuguese, even though still small if compared to the resources that may be available for other languages.

Besides using a larger corpus, we also attempt to overcome data sparseness by using so-called Factored Language Models (FLMs), cf. (Bilmes and Kirchhoff, 2003). FLMs generalise the notion of n-gram models by taking into account, besides word counts, any other source of information (or factors) deemed relevant, such as gender, number, POS etc. When taking only the word (W) factor into account, an FLM may be set to behave exactly like an ordinary n-gram model. FLMs are commonly seen in the speech research community, and have been successfully applied to a number of NLG tasks in high-inflected language generation, e.g., (Novais et al., 2011).

The corpus was part-of-speech tagged using MXPOST[1] and additional gender and number information was obtained from the Brazilian Portuguese lexicon described in (Muniz et al., 2005). After a number of pilot experiments, the tagged corpus was used for training a number of FLMs conveying up to three factors: words (W), lemmas (L) and part-of-speech (P). However, given the amount of data to be processed and limitations in both hardware and software, only FLMs of order 2 and 3 are presently discussed, called 2WL, 3WL, 2WLP and 3WLP models[2]. These models are defined as follows:

$$2WL : p(W_t|W_{t-1}, L_{t-1}).$$
$$3WL : p(W_t|W_{t-1}, L_{t-1}, W_{t-2}, L_{t-2}).$$
$$2WLP : p(W_t|W_{t-1}, L_{t-1}, P_{t-1}).$$
$$3WLP : p(W_t|W_{t-1}, L_{t-1}, P_{t-1}, W_{t-2}, L_{t-2}, P_{t-2}).$$

Likewise an ordinary bigram model, the first model (2WL) considers word (W) frequencies in the first place. However, in case the given word is not found, the 2WL model will take the corresponding lemma (L) into account if this information is available at all. The 3WL is similar, but taking the two previous words/lemmas into account (i.e., as in a trigram model). The 2WLP model is similar to 2WL, but goes one step further by considering - as a last resort - part-of-speech (P) information as well, and 3WLP is once again the trigram alternative. All models use Kneser-Ney smoothing when applicable. For details on how FLMs are built and the related issue of parallel back-off, see (Bilmes and Kirchhoff, 2003).

4. Evaluation

A preliminary evaluation work of our systems was carried out as follows. First, we collected a test corpus of 4,297 on-line newspaper headlines of up to 9 words in length. As in the case of the training corpus, the test corpus was

[1] www.inf.ed.ac.uk/resources/nlp/local_doc/MXPOST.html

[2] Earlier work in (Novais et al., 2011) also described a number of experiments with FLMs that take gender and number factors into account, which have been presently abandoned for similar reasons.

also tagged with POS, gender and number information. In addition to that, the tags in the test corpus were manually verified to ensure correctness. This was particularly necessary in the case of proper names, which occur in large numbers in the newspapers domain, and were often incorrectly tagged as nouns etc.

In order to minimize the possible effects of incorrect tagging, all errors identified in the test corpus were subsequently corrected in the training corpus as well, and the language models described in the previous section were regenerated using the corrected data.

Each sentence in the revised test data set has undergone a process of abstraction (likewise Example 1 in section 2) in which every content word was replaced by its lemma, and sentence structure was represented in form of dependency trees with random linear order of NP and VP constituents. Existing features such as gender, number etc. were preserved when available from the tagged test corpus.

Although the main focus of our evaluation work is to compare different statistical models among themselves, it is of course interesting to compare these models to a non-statistical approach as well. To this end, we developed an additional baseline system - hereby called Rule-based - that makes use of agreement and linearization rules for Brazilian Portuguese to model NP and VP constituency ordering. The Rule-based system is a surface realisation engine for the Brazilian Portuguese language, in many ways not unlike the SimpleNLG system (Gatt and Reiter, 2009) developed for the English language. Full details of the Rule-based system are to be described elsewhere. Briefly, the system makes use of basic grammar rules and the dictionary implemented in (Muniz et al., 2005) to generate the most typcal structures of the Portuguese language (e.g., verb and noun phrases, sentences in active and passive voice etc.) from the same input specification in previous Example 1. As in the case of the statistical methods, the Rule-based system assumes that lexical choice has already been performed, and focus on the agreement and the linearization tasks.

The Rule-based system consists of a collection of JAVA methods to incrementally build sentence constituents and then entire sentences. The following is a simplified example showing a sequence of calls to realise the noun phrase np_1 as 'the Indian writer Amitav Ghosh', which consist of a NP head (Amitav Ghosh) and two pre-modifers m_1 (Indian) and n_1 (writer). The exact order of the modifiers is left to be decided by the system.

NounPhrase np_1 = new def();
np_1.setLemma("Amitav Ghosh");
Modifier m_1 = new Modifier();
m_1.setAdjective("Indian");
Modifier n_1 = new Modifier();
n_1.setProperName("writer");
np_1.setAdjectiveModifier(m_1);
np_1.setProperNameModifier(n_1);

The 4,297 abstract sentence specifications in the test corpus were taken as the input to the four versions of the statistical generator (2WL, 3WL, 2WLP and 3WLP) described in the previous section, and also to the Rule-based system. In the case of the statistical models, over-generation produced (4 systems * 37,462 alternative output strings each) = 149,848 candidate sentences (or an average of 8 alternatives for each input specification, out of which the single most likely alternative is to be selected[3].) The Rule-based system produced one output string for each input, making 4,297 output strings in total.

Evaluation proper was carried out by comparing each of the (5 systems * 4,297 sentences) = 21,485 output sentences to the original version in the test corpus while computing BLEU, NIST, Edit-distance and Accuracy (i.e., exact string match) scores. The results are summarised in Table 1. Recall that higher BLEU, NIST and Accuracy scores - but lower Edit-distance - represent closer proximity to the target corpus.

Strategy	Accuracy	Edit-dist.	BLEU	NIST
Rule-based	0.69	4.55	0.85	14.97
2WL	0.84	2.69	0.93	14.99
3WL	0.86	2.34	0.94	15.02
2WLP	0.88	2.07	0.94	15.04
3WLP	0.89	1.84	0.95	15.06

Table 1: Results

Results for one-way ANOVA comparing edit-distance values[4] (4,297 instances for each system) followed by a Tukey HSD test ($\alpha = 0.01$) showed highly significant differences between the systems (F(4,21480)=118.03, MSE=42.67, $p<0.01$). The homogeneous subsets found are shown in Table 2.

Strategy					
3WLP	A	B			
2WLP	A	B	C		
3WL		B	C	D	
2WL			C	D	
Rule-based					E

Table 2: Homogeneous subsets for Edit Distance scores. Systems which do not share a letter are significantly different at $\alpha = 0.01$.

5. Related Work

The literature in the field presents a large number of generate-and-select approaches to language generation, starting with works such as (Langkilde, 2000; Oh and Rudnicky, 2000; Ratnaparkhi, 2000) and many others. As pointed out in (Gatt and Reiter, 2009), most of the existing surface realisation systems tend to perform two rela-

[3]Given the small scale of the evaluation work, the system simply performs exhaustive search over all potential candidate sentences.

[4]Since BLEU and NIST scores are computed for each system as a whole, and Accuracy is simply a binary value representing each string pair comparison, edit-distance was considered to be the most suitable metric for this analysis.

tively independent tasks: a more domain-dependant mapping from the application semantics onto linguistic forms (including, for instance, the lexical choice task), and a more language-oriented task of sentence linearization.

Likewise the SimpleNLG system presented in (Gatt and Reiter, 2009), our current system is more suitably described as a surface realisation engine, that is, we assume that all lexical choices and other domain-dependent decisions have already been made, and focus on agreement and sentence linearization issues. Differently from SimpleNLG, however, we do not implement realisation rules of any kind, relying entirely on statistical filters. To some extent, our baseline system is closer to SimpleNLG, but once again the system architecture is different, as in the present implementation it was necessary to rely on dictionary information to support the more complex Portuguese morphology.

Our current work builds on a series of previous experiments on individual lexicalization and surface realisation subtasks using n-gram and factored language models alike: the experiments in (Novais et al., 2010) used n-grams models to address the issues of Portuguese NP and VP lexical choice, ordering of NP modifiers, and verb-complement agreement in active and passive voice. Among these issues, the more problematic VP lexical choice, ordering of noun modifiers and verb-complement agreement in passive voice were revisited in (Novais et al., 2011), this time making use of factored language models.

Also in (Novais et al., 2011) a preliminary version of the present system was first sketched. The current work is however our first attempt to generate complete, real sentences using a fully implemented version of the system, and making use of a much larger, purpose-built training corpus.

6. Discussion

Our initial evaluation work suggests that FLMs outperform the non-statistical baseline system, and that the more complex FLMs (2WLP and 3WLP, which take into account word, lemma and part-of-speech factors) are best of all.

One important limitation of the current work is that all possible orderings of NP and VP constituents are evaluated by the language models, including not only more standard NP structures (e.g., NPs such as 'the cover of the magazine', 'the magazine of the cover'[5] etc.) but also ungrammatical structures (e.g., NPs starting by a preposition, as in 'the of cover magazine' etc.) While this strategy is currently adopted to provide maximum language-independency and minimal development costs, the addition of simple (but possibly more language-dependent) rules to prevent some of these illegal candidate structures is likely to have a great impact on the number of alternatives under consideration and, as a result, increase the overall accuracy of the system. In a preliminary study of the sentences in our test corpus, we estimate that by implementing basic rules to avoid the most inconsistent NP syntactic structures our current set of 149,848 candidate sentences could be reduced to less than 20,000 alternatives, that is, achieving some 87% reduction on the number of alternatives under consideration.

[5]In which the second example is of course less likely to be selected than the first one.

Although many - or perhaps most - candidate sentences that would be discarded in this way are unlikely to be selected by the statistical models in the first place, it is still tempting to implement additional rules and examine their impact on the overall accuracy.

Thus, as future work we also intend to add NP constituency rules to the system, and re-evaluate the present strategies using a more limited number of candidate sentences. In doing so, we expect to reach performance levels that are closer to what is required for real-world applications, at the cost of losing some language-independency of the more purely statistical approach discussed in this paper.

Finally, in this paper we have focused on FLMs of order 2 and 3 that use only word, lemma and POS information, and we no longer address the use of FLMs of order 4 or above, or those that take gender and number factors into account. Although this limitation may suggest room for improvement, these simpler models were chosen for practical reasons, as our previous work seems to suggest that these models provide a more suitable balance between accuracy and computational efficiency for the surface realisation task.

7. Acknowledgements

This work has been supported by FAPESP grant nr. 2009/08499-9 and nr. 2009/09061-7, and by the University of São Paulo.

8. References

A. Belz, M. White, D. Espinosa, E. Kow, D. Hogan, and A. Stent. 2011. The first surface realisation shared task: Overview and evaluation results. In *Proceedings of the 13th European Workshop on Natural Language Generation*, pages 217–226.

J. Bilmes and K. Kirchhoff. 2003. Factored language models and generalized parallel backoff. In *Proceedings of the HLT-NAACL 2003 conference, vol.2.*

A. Gatt and E. Reiter. 2009. Simplenlg: A realization engine for practical applications. In *Proceedings of the 12th European Workshop on Natural Language Generation (ENLG-2009)*, Athens.

I. Langkilde. 2000. Forest-based statistical sentence generation. In *Proceedings of ANLPNAACL'00*, pages 170–177.

R. Malouf. 2000. The order of prenominal adjectives in natural language generation. In *Proceedings of the ACL-2000 Conference*, Hong Kong.

M. Mitchell. 2009. Class-based ordering of prenominal modifiers. In *Proceedings of the 12th European Workshop on Natural Language Generation (ENLG-2009)*, pages 50–57, Athens.

M. C. Muniz, E. Laporte, and M. G. V. Nunes. 2005. Unitex-pb, a set of flexible language resources for brazilian portuguese. In *Proceedings of the III Information and Language Technology Workshop (TIL 2005)*.

E. M. Novais, T. D. Tadeu, and I. Paraboni. 2010. Improved text generation using n-gram statistics. *Lecture Notes in Artificial Intelligence*, 6433:316–325.

E. M. Novais, I. Paraboni, and D. T. Ferreira. 2011. Highly-inflected language generation using factored language models. *Lecture Notes in Computer Science*, 6608:429–438.

M. G. V. Nunes, F. Vieira, C. Zavaglia, C. Sossolete, and J. Hernandez. 1996. A construcão de um léxico para o portugues do brasil: licões aprendidas e perspectivas. In *Proceedings of the II Encontro para o processamento de portugues escrito e Falado (PROPOR)*, pages 61–70, Curitiba, Brazil.

A. Oh and A. Rudnicky. 2000. Stochastic language generation for spoken dialogue systems. In *Proceedings of the ANLP-NAACL 2000 Workshop on Conversational Systems*, pages 27–32.

A. Ratnaparkhi. 2000. Trainable methods for surface natural language generation. In *Proceedings of ANLP-NAACL'00*, pages 194–201.

E. Reiter. 2007. An architecture for data-to-text systems. In *Proceedings of the European Natural Language Generation workshop*, pages 97–104.