

Geração de Expressões de Referência usando Relações Espaciais

Diego dos Santos Silva, Ivandré Paraboni

¹School of Arts, Sciences and Humanities (EACH)
University of São Paulo (USP)
Av. Arlindo Bettio, 1000 - São Paulo, Brasil

diego.silva, ivandre@usp.br

Abstract. *Instruction-giving systems for virtual interactive 3D worlds have a wide range of applications in education, games and others. This paper discusses the computational task of referring expression generation for systems of this kind, focusing on the use of spatial relations to describe domain objects.*

Resumo. *Sistemas de geração de instruções em mundos virtuais interativos 3D possuem uma ampla gama de aplicações em áreas como educação, desenvolvimento de jogos etc. Neste artigo discutimos o problema computacional da geração de expressões de referência em ambientes deste tipo, enfocando a questão do uso de relações espaciais para descrever objetos do domínio.*

1. Introdução

A geração de expressões de referência (GER) é um dos componentes fundamentais de aplicações de geração de língua natural (GLN) a partir de dados de entrada não linguísticos. Algoritmos de GER tratam da tarefa de seleção do conteúdo a ser realizado, por exemplo, na forma de descrições definidas como ‘o terceiro homem à esquerda, vestido de preto’.¹

O problema computacional de GER contempla, entre outras questões, o desafio de produzir descrições breves e psicologicamente plausíveis. GER é uma ativa linha de pesquisa em GLN, tendo sido inclusive objeto de uma série de competições (ou *shared tasks*) recentes [Belz and Gatt 2007, Gatt et al. 2008, Gatt et al. 2009]. Abordagens existentes, entretanto, tendem a considerar principalmente domínios simplificados e/ou bidimensionais. O problema de referência em domínios físicos mais realistas (e.g., com grande complexidade estrutural, tridimensionalidade etc.) permanece pouco explorado na pesquisa da área, possivelmente em virtude da própria dificuldade em criar bons modelos computacionais deste tipo.

Mais recentemente, entretanto, este cenário começou a mudar com iniciativas como o projeto GIVE (*Generating Instructions in Virtual Environments*) [Koller et al. 2009]. GIVE é uma plataforma para desenvolvimento e teste de sistemas de GLN em mundos virtuais interativos, na qual o sistema encarrega-se de todo gerenciamento do ambiente gráfico e de interatividade, permitindo ao desenvolvedor concentrar-se apenas na tarefa de GLN e produzir rapidamente uma aplicação de teste. A figura 1 ilustra a interface deste sistema.²

¹Não tratando portanto da questão da *interpretação* (e.g., anafórica) de expressões existentes [Paraboni 1997, Cuevas and Paraboni 2008].

²Extraída de <http://www.give-challenge.org/research/>



Figura 1. Exemplo de uma cena do sistema GIVE [Koller et al. 2009].

O ambiente GIVE é composto de objetos manipuláveis do tipo botão, usados para abrir e fechar portas e outras funções de interação com o mundo virtual, e objetos maiores como peças de mobiliário e afins. Estes objetos podem, a critério do desenvolvedor de GLN, ser empregados como ponto de referência nas instruções fornecidas ao usuário, como em ‘Aperte o botão *ao lado da porta*’. O uso de pontos de referência não é, entretanto um recurso nativo do sistema, e sua implementação requer a extração e manipulação de relações espaciais (e.g., ‘ao lado de’) por parte do algoritmo de GER.

Este artigo discute o desenvolvimento de um algoritmo de GER que faz uso de relações espaciais em ambientes do tipo GIVE. Será tratada especificamente a questão da seleção de conteúdo semântico, deixando de lado o problema da realização textual destas expressões [Pereira and Paraboni 2007, Pereira and Paraboni 2008, de Novais and Paraboni 2012].

2. Trabalhos relacionados

Um dos algoritmos mais conhecidos na área, e que ajudou a definir o próprio problema computacional de GER, é o algoritmo Incremental apresentado em [Dale and Reiter 1995]. Este algoritmo recebe como entrada um contexto C formado por um grupo de objetos, o objeto-alvo ou referente r que se deseja descrever, e suas propriedades semânticas na forma de pares (*atributo, valor*), como em (*cor, azul*).

O objetivo do algoritmo é produzir um conjunto L de propriedades de r tal que L seja capaz de distinguir r de todos os outros objetos em C . As propriedades são incluídas em L incrementalmente (de onde provem o nome do algoritmo) segundo uma ordem P predefinida, desde que contribuam para a desambiguação do referente (i.e., excluindo pelo menos um objeto do contexto C). O algoritmo termina quando um conjunto único (i.e., livre de ambiguidade) de propriedades seja produzido (caso em que L poderia ser realizada, por exemplo, como uma descrição definida), ou até que todas propriedades possíveis em P tenham sido consideradas (caso em que L permaneceria ambígua e poderia ser realizada, por exemplo, como uma descrição indefinida).

Considere o exemplo a seguir, ilustrando um contexto composto por três objetos: dois cachorros (um pequeno e outro grande, tendo o cachorro grande a cor branca e o

cachorro pequeno a cor preta), e um gato de tamanho pequeno e cor preta.

- Obj1: (*tipo*, cachorro), (*tamanho*, pequeno), (*cor*, preto).
- Obj2: (*tipo*, cachorro), (*tamanho*, grande), (*cor*, branco).
- Obj3: (*tipo*, gato), (*tamanho*, pequeno), (*cor*, preto).

Seja o objeto-alvo $r = Obj1$, o contexto $C = \{Obj2, Obj3\}$, e a ordem preferencial de propriedades a serem consideradas $P = \{tipo, cor, tamanho\}$. O algoritmo inicia com uma lista L vazia e percorre P na ordem estabelecida, inserindo em L cada propriedade que exclua pelo menos um objeto em C . Neste exemplo, o algoritmo escolhe inicialmente a propriedade (*tipo*, cachorro) por excluir $Obj3$, que é *tipo* gato. A seguir (na ordem em P), a propriedade (*cor*, preto) exclui $Obj2$, que é de *cor* branca. Como o contexto C não possui mais elementos, o algoritmo retorna a expressão L , que poderia ser realizada, por exemplo, como ‘o cachorro preto’.

A ordem preferencial de seleção de atributos da lista P tem grande impacto sobre o tipo de expressão produzida pelo algoritmo. Por exemplo, se fosse considerada a ordem $P = \{tipo, tamanho, cor\}$ para o domínio acima, a mesma referência a $Obj1$ seria ‘o cachorro pequeno’. Em obediência à máxima de brevidade de Grice [Grice 1975], algoritmos de GER tendem a favorecer a seleção de atributos discriminatórios. Em anos recentes, no entanto, passaram a ser consideradas também questões como a naturalidade da expressão (ou *humanlikeness* em [Belz and Gatt 2007]), dentre muitos outros objetivos desejáveis. Uma visão geral da área de GER e seus principais desafios é apresentada em [Krahmer and van Deemter 2012].

Na proposta original, o algoritmo Incremental manipula apenas propriedades atômicas, mas com adaptações a serem discutidas na seção 4.2 pode também ser aplicado ao caso de propriedades relacionais. Esta modificação é de especial importância para o uso de relações espaciais de que trata este trabalho, como em (*acima,o*), no qual o é um objeto usado como ponto de referência para a descrição do objeto-alvo r .

Alguns sistemas participantes da série de competições *GIVE Challenge* [Byron et al. 2009, Koller et al. 2010, Striegnitz et al. 2011] implementam certos recursos de manipulação de relações espaciais, ainda que de forma pouco documentada (e.g., [Braunias et al. 2010, Schutte and Dethlefs 2010, Garoufi and Koller 2011, Akkersdijk et al. 2011]). Entretanto, como estes sistemas foram avaliados apenas de forma extrínseca (i.e., medindo-se o desempenho global de usuários GIVE na tarefa de navegação) não é possível distinguir o eventual impacto do uso de relações espaciais das outras funcionalidades de cada sistema, as quais incluem, por exemplo, um grande número de melhorias não relacionadas à tarefa de GER.

3. Extração e preparação de dados

O algoritmo proposto neste trabalho faz uso de dados de treinamento extraídos do corpus GIVE-2 [Gargett et al. 2010]. A preparação deste conjunto de dados, bem como do conjunto de teste considerado na avaliação da proposta (seção 5), são discutidas a seguir.

3.1. Extração de atributos espaciais do domínio

Objetos em um ambiente GIVE [Koller et al. 2009] possuem apenas uma propriedade atômica básica representando seu *tipo* (botões, portas, cadeiras etc.) e, no caso dos botões,

uma propriedade *cor*. O primeiro passo deste trabalho foi assim a implementação de um conjunto de métodos básicos para computar atributos espaciais de diversos tipos em ambientes do tipo GIVE [Koller et al. 2009].

Os atributos relacionais computados para um dado objeto-alvo *r* e ponto de referência *o* são: *acima(o)*, *abaixo(o)*, *esquerda(o)*, *direita(o)*, *frente(o)* e *atrás(o)*. Para extração destas relações, foram utilizadas as funções propostas em [Kelleher and Costello 2009], baseadas na posição angular de um objeto em relação ao outro no plano cartesiano.

O algoritmo de extração de atributos espaciais utiliza uma constante de distância máxima *k* única para cada tipo de relação. A posição física das entidades no ambiente GIVE é definida pela coordenada de seu ponto central, e assim entidades maiores como sofás, portas etc. possuem um ponto central mais distante das bordas. Uma entidade está próxima de outra entidade se a diferença entre os valores para os eixos *x*, *y* e *z* é no máximo *k*. Se esta condição for verdadeira, considera-se que há uma relação espacial válida para fins de referência.

Tendo em vista o propósito de gerar expressões de referência livres de ambiguidade, a questão da transitividade destas relações foi aqui desconsiderada. Assim, relações como *esquerda(o)* devem ser entendidas como ‘imediatamente à esquerda’ do objeto *o*, e não contemplando objetos mais distantes que também possam estar à esquerda de *o*.

3.2. Preparação dos dados do corpus GIVE-2

Para treinamento e teste da solução proposta, foi realizada uma tarefa de preparação dos dados do corpus GIVE-2 [Gargett et al. 2010], construído por meio de experimentos envolvendo 36 pares de participantes de língua inglesa e alemã alternando-se nas tarefas de instrutor e jogador. O corpus contém todas instruções fornecidas pelo instrutor, e as respectivas decisões tomadas pelo jogador (e.g., movimentos, ações de pressionar botões etc.). Este conjunto de dados multimodal pode ser visualizado na forma de animação com uso da ferramenta *Replay* apresentada em [Gargett et al. 2010].

Inicialmente, foi realizada a extração do conjunto de objetos e propriedades semânticas de cada um dos três mundos GIVE do corpus. A seguir, as instruções existentes foram utilizadas para extração do conjunto de expressões de referência de interesse e suas propriedades atômicas e relacionais. As expressões aqui consideradas são aquelas que fazem referência a objetos do tipo botão, que são frequentemente referenciados nas instruções de navegação por serem os únicos elementos manipuláveis no mundo GIVE.

Foram extraídas do corpus todas instruções contendo a palavra ‘button’ e formas equivalentes como ‘this’ e ‘box’ inferidas pelo uso de verbos como ‘press’ ou ‘click’. Como no entanto este procedimento não foi exaustivo, é possível que uma pequena parcela de descrições menos comuns tenha sido deixada de fora da presente análise. No total, foram identificadas 992 descrições de interesse. No caso da porção em alemão do corpus, as instruções foram previamente traduzidas para o inglês com uso da ferramenta *Google Translate*³ de modo a facilitar sua interpretação.

A tarefa de preparação de dados produziu um conjunto de descrições anotadas com informações sobre seus atributos atômicos e relacionais, bem como suas informações con-

³<http://translate.google.com.br/>

textuais (i.e., o objeto-alvo e demais objetos do contexto, seus atributos atômicos e relacionais). Este conjunto foi então dividido em um conjunto de treinamento (794 instâncias) e teste (198 instâncias) selecionadas aleatoriamente. O uso destes conjuntos é descrito nas seções a seguir.

4. A abordagem proposta

Nesta seção propomos um algoritmo de GER que faz uso de relações espaciais para descrever objetos do domínio GIVE [Koller et al. 2009]. A proposta é dividida em duas etapas: a tarefa de seleção de uma relação espacial adequada a partir do contexto, e o algoritmo de GER propriamente dito.

4.1. Seleção de pontos de referência

O primeiro módulo da presente proposta é responsável por determinar, para um dado contexto, quais os objetos que mantêm relações passíveis de referência entre si. Foi utilizada para este fim uma abordagem de aprendizagem de máquina semelhante à adotada em [Viethen 2010], porém descartando-se atributos que não se aplicam ao domínio GIVE⁴, e acrescentando-se outros que capturam aspectos específicos da situação a ser tratada.

Dado um objeto-alvo r e um candidato a ponto de referência o , utilizamos um classificador binário *use_relation* para determinar se r pode ser referenciado via o através de uma relação espacial. Os atributos de aprendizagem considerados foram extraídos da porção de treinamento do corpus, e são sumarizados na tabela 1.

Tabela 1. Atributos de aprendizagem para a classe binária *use_relation*

Atributo	Descrição
distractors	Quantidade de objetos do mesmo tipo que o alvo.
landmarks	Quantidade de objetos que mantêm relações espaciais com o alvo.
ambiguous_objects	Quantidade de objetos iguais ao alvo e na mesma sala.
distance	Distância entre o alvo e o ponto de referência.
visible_objects	Quantidade de objetos visíveis no momento da referência.
relation_unique	Indica se a relação entre o alvo e o ponto de referência é única no contexto.
relation_vertical	Indica se a relação entre o alvo e o ponto de referência é vertical.
most_salient_landmark	Indica se o ponto de referência é mais saliente que o alvo.
equal_landmarks	Indica se o ponto de referência é o do mesmo tipo e cor que o alvo.

As instâncias de aprendizagem para a classe *use_relation* foram geradas da seguinte forma. Para cada objeto-alvo r citado no corpus, foi computada uma lista de n objetos que seriam candidatos a pontos de referência em uma possível descrição de r naquele contexto, utilizando-se o método de extração de relações espaciais do domínio descrito na seção 3.1.

A seguir, para cada par alvo-candidato $(r, o_{i=1..n})$, foi gerada uma instância de aprendizagem, totalizando assim n instâncias para cada objeto-alvo. O conjunto de n instâncias de cada objeto-alvo r foi rotulado da seguinte forma: se a descrição de r no corpus não usa uma relação espacial, então todas n instâncias são rotuladas como negativas (*use_relation = no*), isso é, nenhum dos candidatos pode ser recomendado como ponto

⁴Por exemplo, no ambiente GIVE todos objetos de um mesmo tipo possuem o mesmo tamanho, o que torna pouco útil a definição de atributos de aprendizagem baseados nessa característica.

de referência para r . Por outro lado, se a descrição do objeto r no corpus usou uma relação com um ponto de referência o , então a instância que representa o par (r,o) é rotulada como positiva (*use_relation = yes*), e todas outras $n-1$ instâncias como negativas.

Como resultado deste procedimento, foram geradas 3246 instâncias de aprendizagem compostas pelos atributos acima, sendo 335 positivas e 2911 negativas. A classificação propriamente dita foi realizada com o algoritmo de indução de árvores de decisão *J48* disponibilizado pelo pacote WEKA [Witten et al. 2011], utilizando-se *10-fold cross-validation* e demais parâmetros *default* do algoritmo (confiança $C=0,25$ e mínimo de $M=2$ instâncias por folha).

Os resultados médios obtidos pela indução deste classificador são $F=0.66$ para classe positiva, e $F=0.96$ para a classe negativa, o que consideramos satisfatório dado que este classificador objetiva modelar apenas o comportamento *médio* dos 72 participantes do experimento que deu origem ao corpus GIVE-2, os quais frequentemente adotam estratégias de referência conflitantes. Por exemplo, um mesmo botão pode ser descrito por participantes distintos como ‘o botão azul’ ou ‘o botão ao lado da porta’, dentre muitas outras possibilidades observadas no corpus.

O classificador foi incorporado ao ambiente GIVE e constitui o primeiro módulo a ser invocado na produção de uma descrição de um objeto-alvo r . Inicialmente, são computadas todas relações existentes entre as entidades do contexto em que r se encontra. Com base nestas informações, é gerada uma série de n instâncias de teste como descrito acima, as quais são submetidas ao classificador. Dentre as relações classificadas como positivas, aquela que tenha sido observada com maior frequência na porção de treinamento do corpus será considerada para uso pelo algoritmo de GER descrito a seguir.

4.2. Geração de expressões de referência usando relações espaciais

Utilizando-se o procedimento descrito na seção anterior é possível determinar, para uma determinada situação de referência a um objeto-alvo r , qual relação espacial seria mais adequada para descrever r naquele contexto. A etapa seguinte consiste então da geração da descrição propriamente dita, a qual pode ou não incluir a relação espacial sugerida.

O algoritmo proposto para este fim é uma versão modificada do algoritmo Incremental [Dale and Reiter 1995] para manipular propriedades relacionais, e também integrado ao sistema GIVE [Koller et al. 2009]. Nesta proposta, as propriedades atômicas e relacionais existentes são consideradas para inclusão em ordem de frequência, conforme observado na porção de treinamento do corpus GIVE-2. Uma propriedade é incluída na expressão resultante desde que elimine ao menos uma entidade do contexto.

No caso de expressões envolvendo um objeto e um ponto de referência, como em ‘o livro sobre a mesa’, o algoritmo descreve cada objeto de forma independente, o que pode acarretar sobre-especificação. Por exemplo, em um contexto com duas mesas e dois livros, em que um dos livros está sobre uma das mesas, o presente algoritmo faria uso de propriedades adicionais, como em ‘o livro vermelho, sobre a mesa da esquerda’. Esta medida foi adotada para evitar possíveis problemas de identificação de objetos em domínios espaciais complexos, como os discutidos em [Paraboni 2003, Paraboni et al. 2006, Paraboni and van Deemter 2013].⁵

⁵Diferentemente de [Dale and Haddock 1991], por exemplo, que neste caso permitiria mútua

5. Avaliação

A avaliação da solução proposta foi realizada de forma intrínseca com base no conjunto de teste descrito na seção 3.2, considerando-se três aspectos de interesse: a política de seleção da relação espacial, a ordem de preferência para seleção de propriedades (o parâmetro P do algoritmo de GER), e o tratamento de relações espaciais redundantes.

Com relação à política de seleção da relação espacial, consideramos duas alternativas: a proposta original - que seleciona a relação espacial *mais frequente* no corpus de treinamento - e uma alternativa na qual a propriedade espacial é selecionada de forma *aleatória* dentre as opções disponíveis.

Com relação à ordem de preferência P utilizada pelo algoritmo, consideramos também duas alternativas: a proposta original - que faz ordenação *por frequência* conforme observado no conjunto de treinamento - e uma estratégia *gulosa* na qual propriedades de maior poder discriminatório (ou seja, aquelas que diferenciam o objeto-alvo do maior número possível de objetos do contexto) têm preferência.

Finalmente, com relação ao tratamento de relações espaciais redundantes, consideramos duas alternativas: a proposta original - que *não* inclui propriedades redundantes (i.e., aquelas que não contribuem para a desambiguação do referente, como em [Dale and Reiter 1995]) - e uma estratégia na qual a relação espacial é selecionada mesmo que seja *redundante*.

A avaliação destes três aspectos da solução leva ao enunciado de ($2 * 2 * 2 = 8$) algoritmos distintos, dos quais a proposta original corresponde à alternativa que seleciona a relação espacial mais frequente, ordena a lista de preferências P também por frequência, e inclui uma propriedade relacional na expressão apenas se esta for discriminatória.

Por simplicidade, todas alternativas avaliadas consideram como contexto de referência o conjunto de objetos na mesma sala onde se encontra o objeto-alvo. Entretanto, cabe observar que, no caso do corpus GIVE-2, o contexto utilizado pelos participantes do experimento foi, em alguns casos, formado apenas pelos objetos visíveis naquele instante. Por exemplo, quando o jogador já estava muito próximo do objeto referenciado pelo instrutor, em alguns casos este optou por desconsiderar os objetos mais distantes e produzir uma descrição breve como ‘o botão’, mesmo havendo outros botões na mesma sala. Uma vez que estes casos não foram contemplados na presente avaliação, assume-se assim um ônus para todos os algoritmos avaliados.

Cada uma das 198 situações de referência constantes no corpus de teste - aqui denominado conjunto *Referência* - foi fornecida como entrada para cada um dos 8 algoritmos, resultando assim em 8 conjuntos de expressões de referência aqui denominados conjuntos *Sistema 1..8*. A avaliação propriamente dita consistiu em comparar cada conjunto *Sistema 1..8* com o conjunto *Referência*.

Para a comparação de cada um dos ($8 * 198 = 1584$) pares *Sistema-Referência*, utilizamos duas métricas amplamente empregadas em trabalhos da área como [Belz and Gatt 2007, de Lucena et al. 2010]: o coeficiente *Dice* [Dice 1945], que mede o grau de similaridade entre os dois conjuntos de propriedades, assumindo um valor entre 0 (totalmente distintos) e 1 (idênticos); e *MASI* [Passonneau 2006], que possui correlação desambiguação como em ‘o livro sobre a mesa’.

com *Dice*, porém atribuindo maior peso no caso de uma expressão ser subconjunto da outra. Os 8 algoritmos e seus resultados são sumarizados na tabela 2. O algoritmo originalmente proposto é o primeiro (#1).

Tabela 2. Resultados

#	Seleção	Ordenação	Redundância	Dice	MASI
1	mais frequente	por frequência	não	0.73	0.49
2	mais frequente	por frequência	sim	0.58	0.22
3	mais frequente	gulosa	não	0.28	0.11
4	mais frequente	gulosa	sim	0.21	0.05
5	aleatória	por frequência	não	0.73	0.49
6	aleatória	por frequência	sim	0.58	0.22
7	aleatória	gulosa	não	0.25	0.10
8	aleatória	gulosa	sim	0.20	0.05

As seguintes comparações entre algoritmos foram realizadas utilizando-se o teste de *Wilcoxon* [Wilcoxon 1945] sobre coeficientes *Dice*. Primeiramente, observa-se que não houve diferença significativa entre a seleção da propriedade mais frequente e a simples seleção aleatória, o que se explica pelo fato de que nos dados de teste a maioria dos objetos-alvo só possuir um ponto de referência possível.

Em segundo lugar, a ordenação por frequência é significativamente superior à ordenação gulosa para todos os pares de algoritmos avaliados (i.e., comparando-se #1 com #3, #2 com #4 etc.) ($W=16110$, $Z=11,6$, $p < 0,001$).

Além disso, a estratégia de não inserir redundância é significativamente superior ao seu uso para todos os pares de algoritmos (i.e., comparando-se #1 com #2, #3 com #4 etc.). A menor diferença observada, mas ainda altamente significativa, foi entre os algoritmos #7 e #8 ($W=986$, $Z=5,75$, $p < 0,001$).

6. Discussão

Este artigo descreveu o desenvolvimento e avaliação de um algoritmo de GER que faz uso de relações espaciais em ambientes do tipo GIVE [Koller et al. 2009]. Apesar da relativa simplicidade da proposta, este estudo contribui para o entendimento de três aspectos do problema: a política de seleção de propriedades espaciais, a ordenação das propriedades consideradas pelo algoritmo de GER, e o tratamento de propriedades redundantes.

Mesmo não tendo sido observada uma diferença significativa entre as alternativas de seleção de propriedade espacial, as estratégias de ordenação baseada em frequência e de inclusão de propriedades espaciais discriminatórias revelaram-se superiores às demais. Em outras palavras, não parece haver uma preferência geral por propriedades discriminatórias, mas as propriedades espaciais são empregadas principalmente desta forma.

Como trabalho futuro planejamos expandir a presente análise para incluir aspectos adicionais da solução, tais como a prioridade da relação espacial considerada pelo algoritmo, e uma política de seleção de relações espaciais mais sofisticada.

Esta pesquisa conta com apoio FAPESP.

Referências

- Akkersdijk, S., Langenbach, M., Loch, F., and Theune, M. (2011). The Thumbs Up! Twente system for GIVE 2.5. In *Generation Challenges Session at ENLG-2011*, pages 312–317.
- Belz, A. and Gatt, A. (2007). The attribute selection for GRE challenge: Overview and evaluation results. In *UCNLG+MT: Language Generation and Machine Translation*.
- Braunias, J., Boltz, U., Drager, M., Fersing, B., and Nikitina, O. (2010). The GIVE-2 challenge: Saarland NLG system. In *INLG-2010*.
- Byron, D., Koller, A., Striegnitz, K., Cassell, J., Dale, R., Moore, J., and Oberlander, J. (2009). Report on the first NLG challenge on generating instructions in virtual environments (GIVE). In *ENLG-2009*.
- Cuevas, R. and Paraboni, I. (2008). A machine learning approach to portuguese pronoun resolution. *Advances in Artificial Intelligence—IBERAMIA 2008*, LNAI 5290:262–271.
- Dale, R. and Haddock, N. J. (1991). Content determination in the generation of referring expressions. *Computational Intelligence*, 7:252–265.
- Dale, R. and Reiter, E. (1995). Computational interpretations of the Gricean maxims in the generation of referring expressions. *Cognitive Science*, 19.
- de Lucena, D. J., Pereira, D. B., and Paraboni, I. (2010). From semantic properties to surface text: The generation of domain object descriptions. *Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial*, 14(45):48–58.
- de Novais, E. M. and Paraboni, I. (2012). Portuguese text generation using factored language models. *Journal of the Brazilian Computer Society*, 19(2):135–146.
- Dice, L. R. (1945). Measures of the amount of ecologic association between species. *Ecology*, 26(3):297–302.
- Gargett, A., Garoufi, K., Koller, A., and Striegnitz, K. (2010). The GIVE-2 corpus of giving instructions in virtual environments. In *LREC-2010*.
- Garoufi, K. and Koller, A. (2011). The Potsdam NLG systems at the GIVE-2.5 challenge. In *Generation Challenges Session at ENLG-2011*, pages 307–311.
- Gatt, A., Belz, A., and Kow, E. (2008). The TUNA challenge 2008: Overview and evaluation results. In *INLG-2008*, pages 198–206.
- Gatt, A., Belz, A., and Kow, E. (2009). The TUNA-REG challenge 2009: Overview and evaluation results. In *ENLG-2009*, pages 174–182.
- Grice, H. P. (1975). Logic and conversation. In Cole, P. and Morgan, J. L., editors, *Syntax and semantics*, volume 3. New York: Academic Press.
- Kelleher, J. D. and Costello, F. J. (2009). Applying computational models of spatial prepositions to visually situated dialog. *Computational Linguistics*, 35(2):271–306.
- Koller, A., Byron, D., Cassell, J., Dale, R., Striegnitz, K., Moore, J., and Oberlander, J. (2009). The software architecture for the first challenge on generating instructions in virtual environments. In *EACL-2009*.

- Koller, A., Striegnitz, K., Gargett, A., Byron, D., Cassell, J., Dale, R., Moore, J., and Oberlander, J. (2010). Report on the second NLG challenge on generating instructions in virtual environments (GIVE-2). In *INLG-2010*.
- Krahmer, E. and van Deemter, K. (2012). Computational generation of referring expressions: A survey. *Computational Linguistics*, 38(1):173–218.
- Paraboni, I. (1997). Uma arquitetura para a resolução de referências pronominais possessivas no processamento de textos em língua portuguesa. Master's thesis, PUCRS.
- Paraboni, I. (2003). *Generating references in hierarchical domains: the case of Document Deixis*. PhD thesis, University of Brighton.
- Paraboni, I., Masthoff, J., and van Deemter, K. (2006). Overspecified reference in hierarchical domains: measuring the benefits for readers. In *INLG-2006*, pages 55–62.
- Paraboni, I. and van Deemter, K. (2013). Reference and the facilitation of search in spatial domains. *Language and Cognitive Processes*.
- Passonneau, R. (2006). Measuring agreement on set-valued items (MASI) for semantic and pragmatic annotation. In *LREC-2006*.
- Pereira, D. B. and Paraboni, I. (2007). A language modelling tool for statistical NLP. In *5th Workshop on Information and Human Language Technology (TIL-2007)*, pages 1679–1688, Rio de Janeiro. Sociedade Brasileira de Computação.
- Pereira, D. B. and Paraboni, I. (2008). Statistical surface realisation of portuguese referring expressions. In *Advances in Natural Language Processing*, volume LNAI 5221, pages 383–392. Springer-Verlag.
- Schutte, N. and Dethlefs, N. (2010). The Dublin-Bremen system for the GIVE-2 challenge. In *INLG-2010*.
- Striegnitz, K., Denis, A., Gargett, A., Garoufi, K., Koller, A., and Theune, M. (2011). Report on the second second challenge on generating instructions in virtual environments (GIVE-2.5). In *Generation Challenges Session at ENLG-2011*, pages 270–279.
- Viethen, H. A. E. (2010). *The Generation of Natural Descriptions: Corpus-based Investigations of Referring Expressions in Visual Domains*. PhD thesis, Macquarie University.
- Wilcoxon, F. (1945). Individual comparisons by ranking methods. *Biometrics Bulletin* 1, 6(2):80–83.
- Witten, I. H., Frank, E., and Hall, M. A. (2011). *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufman Publishers, Burlington, MA, 3rd edition.