Geração de Linguagem Natural para Conversão de Dados em Texto Aplicação a um Assistente de Medicação para o Português

Trainable NLG for Data to Portuguese – With application to a Medication Assistant

José Casimiro Pereira
Instituto Politécnico de Tomar
Portugal
casimiro@ipt.pt

António Teixeira
Dep. Electrónica Telec. & Informática/IEETA
Universidade de Aveiro, Portugal
ajst@ua.pt

Resumo

Novos equipamentos como 'smartphones' ou 'tablets' têm revolucionado a interacção do ser humano com a tecnologia, proporcionando novos desafios e oportunidades. Estes novos dispositivos são multimodais por natureza. De entre as várias modalidades, são particularmente interessantes as relacionadas com a interacção por voz e texto. Para que estas formas de interação possam ser usadas entre sistemas e utilizadores humanos, é essencial a existência de módulos capazes de traduzir as informações internas das aplicações em frases ou textos, para visualização no ecrã ou para serem sintetizados de forma a serem ouvidos. É, também, essencial que estes módulos possam gerar frases e textos nas línguas nativas dos utilizadores; que o processo de desenvolvimento não implique grandes conhecimentos e recursos, incluindo tempo de desenvolvimento; e o resultado da geração apresente a variabilidade necessária.

O objectivo principal é o de propor, implementar e avaliar um método de conversão de Dados-para-português passível de ser desenvolvido com um mínimo de tempo e conhecimentos, mas sem comprometer a indispensável variabilidade e qualidade do que é gerado. O sistema apresentado, desenvolvido para um cenário de assistência à toma de medicamentos, destina-se a criar descrições, em linguagem natural, de informação sobre medicação a tomar. Motivados por resultados recentes, optou-se por uma abordagem baseada em tradução automática, com os modelos treinados num pequeno corpus paralelo.

Para isso, foi criado um novo corpus que, depois de validado, foi utilizado no desenvolvimento do sistema. Foram criadas duas variantes do sistema: uma orientada à tradução baseada em sintagmas e outra fazendo uso de informação sintáctica. Foram realizadas avaliações utilizando métricas automáticas – BLEU e Meteor – bem como avaliações por humanos. Os resultados do sistema orientado a sintagmas foram francamente superiores aos do seu concorrente, obtendo uma média por avaliador humano de 60% de frases consideradas inteligíveis, contra 46% do seu congénere, o que pode considerar-se um bom resultado tendo em conta a dimensão do corpus.

Palavras chave

Geração de linguagem natural, Dados-para-Texto, tradução automática, assistência à toma de medicação

Abstract

New equipments, such as smartphones and tablets, are changing human computer interaction. These devices present several challenges, especially due to their small screen and keyboard. In order to use text and voice in multimodal interaction, it is essential to deploy modules to translate the internal information of the applications into sentences or texts, in order to display it on screen or synthesize it. Also, these modules must generate phrases and texts in the user's native language; the development should not require considerable resources; and the outcome of the generation should achieve a good degree of variability.

Our main objective is to propose, implement and evaluate a method of data conversion to Portuguese which can be developed with a minimum of time and knowledge, but without compromising the necessary variability and quality of what is generated. The developed system, for a Medication Assistant, is intended to create descriptions, in natural language, of medication to be taken. Motivated by recent results, we opted for an approach based on machine translation, with models trained on a small parallel corpus.

For that, a new corpus was created. With it, two variants of the system were trained: phrase-based translation and syntax-based translation. The two variants were evaluated by automatic measurements – BLEU and Meteor – and by humans. The results showed that a phrase-based approach produced better results than a syntax-based one: human evaluators evaluated 60% of phrase-based responses as good, or very good, compared to only 46% of syntax-based responses. Considering the corpus size, we judge this value (60%) as good.

Keywords

Natural Language Generation, NLG, data2text, machine translation, medication assistant

1 Introdução

1.1 Motivação

O aumento do uso de dispositivos móveis, como *Smartphones*, *tablets* ou pequenos computadores, é, hoje, uma realidade indesmentível . Uma das principais dificuldades da sua utilização resulta das reduzidas dimensões do teclado e do ecrã. Estas características constituem ao mesmo tempo um desafio e uma oportunidade para o surgimento de novas tecnologias e interfaces.

Estes novos dispositivos são multimodais por natureza, uma vez que permitem várias formas de interacção: texto, imagens, voz, toque, vibração, etc..

De entre estas várias modalidades, são particularmente interessantes as relacionadas com a interacção por voz e texto. Para que tal seja possível, é essencial a existência de módulos capazes de traduzir as informações internas das aplicações em frases ou textos, para visualização no ecrã ou para serem sintetizadas de forma a serem ouvidas pelo utilizador.

Como requisitos adicionais, mas essenciais, temos:

(1) A necessidade desses módulos gerarem frases e textos com a suficiente variedade/variabilidade ao longo do tempo – uma das importantes características das frases produzidas pelos humanos – para que não se tornem aborrecidos e, em consequência, sejam considerados pouco naturais e utilizáveis;

(2) A possibilidade de criar módulos adequados para um número crescente de aplicações, sem serem necessários grandes conhecimentos de áreas não dominadas pelos *developers* de aplicações em geral(como conhecimentos aprofundados de Linguística), e sem ser necessário um grande investimento em termos de tempo de desenvolvimento;

(3) Utilização da língua portuguesa (quinta língua mais falada no mundo), abrindo portas à utilização deste tipo de aplicações na sua língua nativa a cerca de 240 milhões de pessoas (Alves, 2011), com expectativas de crescimento para perto de 335 milhões, em 2050 (Agência Lusa, 2010).

Muitos têm sido os esforços no sentido de dotar os computadores em geral, e estes dispositivos móveis em particular, da capacidade de "falarem" com os seres humanos (Jurafsky & Martin, 2009). Um dos primeiros esforços foi a criação de interacção através de frases e textos pré-definidos ou utilizando mensagens de voz pré-gravadas. Se, por um lado, a opção por modelos pré-definidos (vulgo *templates*) pode permitir a existência de sistemas simples de uma forma rápida, por outro, sem um investimento grande na criação de um número elevado de *templates*, teremos uma indesejável repetição das frases e textos produzidas pelo sistema. Esta repetição sistemática de uma mesma frase tipo resulta em sistemas percepcionados pelos utilizadores como pouco naturais, reduzindo a sua usabilidade e aceitação por parte destes.

Com o tempo foram propostas formas alternativas de geração de frases e texto, como os sistemas clássicos de Geração automática de Língua Natural (GLN) – em inglês, Natural Language Generation, ou NLG (Reiter & Dale, 1997, 2000). Os sistemas de GLN precisam de mapear alguma fonte de informação (como uma base de dados, por exemplo) em algum tipo de mensagem gerada automaticamente (Bateman & Zock, 2004). No entanto, esta tarefa está longe de ser considerada trivial, necessitando o seu desenvolvimento de muitos recursos (conhecimentos, corpora e tempo). A estes sistemas é requerido que decidam "como" dizer, depois de terem decidido "o que" dizer (Lemon, 2010; Bateman & Zock, 2004). Isto significa que os sistemas de GLN devem imitar os seres humanos, produzindo mensagens que são sintáctica e semanticamente corretas, além de serem, também, contextualmente adequadas.

Apesar de já existirem algumas experiências bem-sucedidas na criação de sistemas (Hunter et al., 2005; Konstantopoulos et al., 2008; McCauley et al., 2008), os recursos necessários para o desenvolvimento de sistemas de GLN clássicos genéricos continuam a ser escassos e o processo moroso, requerendo conhecimentos aprofundados de áreas como a Linguística e Processamento de Linguagem Natural. Para além disso, a sua adaptação a novos requisitos é, em geral, bastante difícil (Lemon, 2010). Constata-se também, através de uma análise da literatura nesta área, que a maioria dos sistemas e recursos necessários foi desenvolvida para a língua inglesa. Em resumo, este tipo de sistemas não se apresenta capaz de cumprir com os requisitos apresentados, sendo necessário explorar alternativas, em especial as que permitam obter sistemas para português e sem um grande investimento em termos de recursos.

Atendendo a que em muitas aplicações concretas para ambientes móveis a parte inicial do problema de geração – "o que" dizer – se encontra resolvido, apenas se torna necessário um sistema mais simples. Esta variante, designada habitualmente por sistemas de conversão de Dados-Para-

Texto (em inglês, Data2Text) (Reiter, 2007), utiliza como fonte para a geração um recurso não linguístico, frequentemente informação interna à aplicação, como dados de alguma fonte de dados.

Conjugando o atrás exposto, o nosso objectivo principal é o de propor, implementar e avaliar um método de conversão de Dados-Para-Português, passível de ser desenvolvido com um mínimo de tempo e conhecimentos, mas sem comprometer a indispensável variabilidade e qualidade do que é gerado (ex: frases).

Para que se possa atingir esse objectivo, com base numa análise de sistemas recentes Data2Text, será explorada a capacidade de sistemas baseados na utilização de aprendizagem automática de uma tradução entre a informação interna e português, tendo por base um corpus paralelo.

1.2 Cenário de aplicação escolhido

Tendo em conta o envelhecimento acentuado da população, o estudo centrou-se neste este grupo de utilizadores. Devido à sua idade, as suas capacidades motoras e cognitivas são mais reduzidas, pelo que a introdução deste tipo de tecnologias, com interfaces em língua natural, oral e escrita, tende a facilitar a sua vida diária. É, também, potenciadora da diminuição do isolamento, da exclusão e do aumento da capacidade de trabalho e autonomia (Teixeira et al., 2013b).

O cenário idealizado refere-se a uma situação onde uma pessoa está a tomar medicamentos. Neste contexto, o sistema deve interagir com o utilizador, usando, como meio de comunicação, a língua portuguesa, assistindo-o nas suas necessidades. Por exemplo, se o utilizador perguntar pelo próximo medicamento a tomar, deverá receber como resposta a informação pretendida, bem como informação complementar que o ajude na sua decisão.

O artigo encontra-se organizado da seguinte forma: uma primeira secção, com uma descrição da motivação e objectivos para a realização deste trabalho, seguida da descrição do cenário da aplicação. Na secção 2, é feita uma breve descrição de alguns exemplos de sistemas de Dados-para-Texto e formas de os avaliar. De seguida, é feita uma descrição da arquitectura geral do sistema implementado. A secção 4 apresenta informação sobre o corpus criado e os desenvolvimentos efectuados para o preparar para estas experiências, seguindo-se, na secção 5, a descrição dos passos efectuados no treino dos diversos sistemas. A secção 6 apresenta os resultados da avaliação e informação sobre as primeiras utilizações. O artigo termina com discussão e conclusões, analisando criticamente os resultados e apontando perspectivas de trabalhos futuros.

2 Trabalho Relacionado

2.1 Exemplos de sistemas de conversão de Dados-para-Texto

Nesta secção, são apresentados alguns exemplos de sistemas orientados a Dados-Para-Texto. Apesar de existirem diversos trabalhos na área da Geração de Língua Natural em português (Oliveira, 2012; Fonseca, 1993; Mendes, 2004; Ribeiro, 1995; Soares, 2001), apenas conseguimos identificar dois estudos relativos ao tema deste subtipo de GLN.

Pollen Forecast for Scotland – é um sistema que pretende traduzir, em texto, uma previsão para a concentração de pólen, nas diversas zonas da Escócia (Turner et al., 2006), de modo a que as pessoas sensíveis a níveis de pólen elevados possam precaver-se. Utiliza um corpus alinhado de 69 pares de frases, correspondentes a níveis de concentração e descrições, escritas por pessoas, referentes a essas concentrações. Este projecto surge como continuação do projecto Sumtime (Hunter et al., 2005), que efectua descrições textuais de previsões de meteorologia, em função dos dados meteorológicos fornecidos.

BabyTalk – Este sistema (Portet et al., 2009; Hunter et al., 2011) surgiu com o objectivo de apoiar os profissionais de saúde (enfermeiros e médicos) de uma Unidade de Cuidados Intensivos Neonatais. Efectivamente, estes profissionais, ao entrarem no seu turno, têm necessidade de assimilar uma grande quantidade de informação, em muito pouco tempo, sobre os bebés aí internados. Essa informação, normalmente, está distribuída por uma grande quantidade de dados sobre os bebés (análises de laboratório, dados dos equipamento de apoio à vida, dados sobre intervenções anteriores, etc.). O BabyTalk proporciona a esses profissionais resumos dos dados relevantes, tornando mais fácil e rápida a assimilação da informação prestada. Como corolário deste projecto, o BabyTalk pretende construir um módulo que permita às famílias terem, em tempo útil, resumos do estado de saúde dos seus bebés (Hunter et al., 2011).

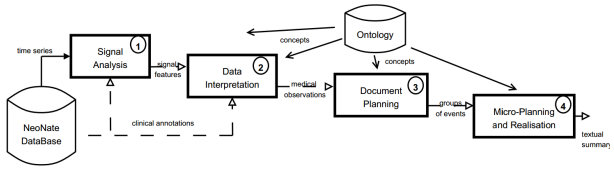


Figura 1: Arquitectura do BabyTalk (BT-45), retirado de (Portet et al., 2009).

SinNotas – É um dos poucos exemplos, de que temos conhecimento, de sistemas de Dados-Para-Texto, orientados para a língua portuguesa. Desenvolvido no Brasil, por Novais e Araújo (de Araújo et al., 2010; Novais et al., 2009), destina-se a dar apoio a uma aplicação de divulgação de notas de alunos, numa universidade. O SinNotas utiliza um corpus alinhado, onde a cada nota possível de um aluno se associa uma descrição para essa nota. Os autores defendem que, com este sistema, conseguiram que os alunos tivessem uma melhor percepção do entendimento dos professores sobre o seu desempenho.

Atributo	Descrição	Valores possíveis / Número de instâncias
provas_aval	Regular exams grades	nao_realizou(50), muito_abaixo(30), razoável(40), bom_mas_baixo(6), bom(84), muito_bom(19), excelente(12)
provas_turma	Same, as compared to the entire class	nulo(50), abaixo(100), acima(91)
progresso	Overall progress throughout the term	nulo(50), declinio(50), menor_meio(48), maior_meio(65), aumento(28)
sub_aval*	Substitutive exams grades	nulo(223), muito_abaixo(16), abaixo(2), acima(0)
sub_turma*	Same, as compared to the entire class	nulo(214), abaixo(11), acima(16)
eps_aval	Practical exercises grades	nao_realizou(56), muito_abaixo(2), razoável(5), bom_mas_baixo(2), bom(22), muito_bom(33), excelente(121)
dev_ep1	Whether exercises were compulsory	nulo(207), sim(34)
freq_aval	Attendance to the lectures	nulo(188), nenhuma(44), insuficiente(9)
corel_nota_falta*	Lower grades / attendance relation	nulo(215), sim(26)
mf_aval	Final term exams	muito_abaixo(81), razoável(41), bom_mas_baixo(5), bom(70), muito_bom(27), excelente(17)
mf_turma	Same, as compared to the entire class	nulo(58), abaixo(48), acima(135)
rec_aval	Recuperation exams grades	nulo(200), muito_abaixo(17), razoável(8), bom_mas_baixo(0), bom(16), muito_bom(0), excelente(0)
aband_rec*	Abandoned recuperation exams	nulo(235), sim(6)
rec_turma	Same, as compared to the entire class	nulo(204), abaixo(16), media(2), acima(19)

Tabela 1: Mensagens e possíveis valores do SINotas (extraído de (Novais et al., 2009)).

PortNLG – É um recente exemplo de um sistema, visando o português como língua de trabalho, desenvolvido por Silva Junior, Paraboni e Novais (Silva Junior et al., 2013). Consiste numa biblioteca JAVA concretizando um realizador superficial. Destina-se a gerar frases em português, tendo como entrada uma especificação abstracta da frase a ser construída.

Mountain – O sistema Mountain foi desenvolvido por Langner(Langner & Black, 2009; Langner, 2010), como parte da sua tese de doutoramento. O Mountain utiliza, também, um corpus alinhado, e foi, dos sistemas analisados, o primeiro a utilizar a ferramenta MOSES (Koehn et al., 2007; Koehn, 2014) como forma de gerar os textos a serem apresentados aos seus utilizadores. A sua 'linguagem de entrada' corresponde a uma sequência de códigos que representam a disponibilidade de um 'court' de ténis. A 'linguagem de saída' corresponde à 'tradução' desse código em inglês.

000000 d5 d3	friday evening is completely closed
100000 d2 t2	the ony time available is noon
111111 d4 t1	the court is open all morning
111111 d1 t3	you can reserve a court anytime on monday evening
100011 d5 t3	six, ten or eleven
010011 d3 t2	you an reserve a court at 1pm, 4pm and 5pm on wednesday
011001 d4 t3	any time but 6, 9 and 10
111011 d7 d2	afternoon except the 3pm block
111100 d1 t2	you can reserve a court is free anytime from noon until 3
110111 d6 t3	saturday evening, ooh, that

Tabela 2: Exemplo do corpus do Mountain (retirado de (Langner, 2010)).

2.2 Avaliação da Geração

Para a utilização efectiva do sistema, é necessário efectuar testes para garantir a sua qualidade. Contudo, a avaliação de sistemas de GLN ainda não é consensual (Hastie & Belz, 2014) e, ao longo dos últimos anos, têm surgido diversas propostas que se podem dividir em dois grandes grupos de avaliação: avaliação feita por seres humanos e avaliações automáticas, efectuadas por computador. Estudos sugerem que as avaliações efectuadas por seres humanos são geralmente melhores do que as automáticas, quando o objecto de estudo são textos de apoio à realização de tarefas, apesar de, em alguns casos particulares, tal possa não se verificar (Law et al., 2005). Apesar desta realidade, as avaliações automáticas têm vindo a ser cada vez mais utilizadas, em especial devido ao enorme custo, em termos de tempo e recursos económicos, que a avaliação por seres humanos acarreta.

Consideram-se, então, para este tipo de sistemas, essencialmente 3 tipos de avaliações: avaliação orientada à tarefa, avaliação por humanos e métricas automáticas. A avaliação orientada à tarefa consiste em produzir os textos e, depois, entregá-los às pessoas que os vão utilizar. O objectivo é avaliar o quanto esses textos ajudam as pessoas a efectuar as suas tarefas. Estas avaliações consomem muito tempo, são bastante dispendiosos e difíceis de concretizar, especialmente quando envolvem pessoas muito qualificadas (Portet et al., 2009). A avaliação por humanos é efectuada fornecendo o texto gerado a uma, ou mais pessoas, e solicitando a

sua avaliação sobre a utilidade e correcção desse texto. As métricas automáticas foram desenvolvidas para substituir as avaliações envolvendo humanos, devido às restrições que esse tipo de avaliação encerra. Este tipo de métricas efectua a comparação entre o texto produzido pelo sistema e texto escrito por humanos, tendo por base a mesma fonte inicial.

BLEU – É o acrónimo para BiLingual Evaluation Understudy (Papineni et al., 2002). O BLEU é um algoritmo que avalia a aproximação entre um texto gerado automaticamente e um texto, previamente obtido, gerado por um ser humano. Quanto mais próximos tiverem, maior qualidade terá o texto em avaliação. Esta avaliação é efectuada sobre os elementos individuais do texto gerado (frases ou partes de frases), comparando-os com um texto de referência, com boa qualidade. O índice desta avaliação é depois extrapolado para todo o texto. Factores como a inteligibilidade ou questões gramaticais não são tidos em consideração nesta métrica. O BLEU é expresso através de um número entre 0 e 1. Quanto maior o valor, maior a similitude com o texto de comparação. Devido à forma como o teste é realizado – comparação entre 'ngrams' –, a avaliação produz valores aceitáveis, quando o texto em avaliação é confrontado com todo o texto de referência e produz valores maus, quando confrontado com simples frases individuais.

Meteor – É o acrónimo de Metric for Evaluation of Translation with Explicit ORdering (Denkowski & Lavie, 2014, 2011). Semelhantemente ao BLEU, o Meteor aplica sobre cada frase gerada o algoritmo de avaliação. Este algoritmo cria um alinhamento entre os constituintes (palavras) da frase em teste e uma frase de referência. Por alinhamento, entende-se uma correspondência direta entre dois *unigrams*, um da frase em análise e outro da frase de referência. A correspondência pode ocorrer ao nível do reconhecimento exato da palavra, se forem sinónimas ou se derivarem de uma mesma palavra. A correspondência pode ocorrer, também, se a frase em avaliação for paráfrase de outra considerada válida.

O índice desta métrica é obtido pelo cálculo da média harmónica entre o número de alinhamentos considerados corretos e o número de alinhamentos possíveis, sendo que este segundo valor tem maior peso que o primeiro. Desenvolvida para minorar alguns dos problemas evidenciados pelo BLEU, esta métrica foca-se nas frases do corpus de forma individual, enquanto o BLEU avalia essencialmente o corpus como um todo.

3 Arquitectura Geral do Sistema

A arquitectura geral do sistema aqui descrito é apresentada na figura seguinte (Figura 2). A sua missão consiste em gerar mensagens em língua natural. É um *componente* de um sistema mais vasto, que se encontra em desenvolvimento.

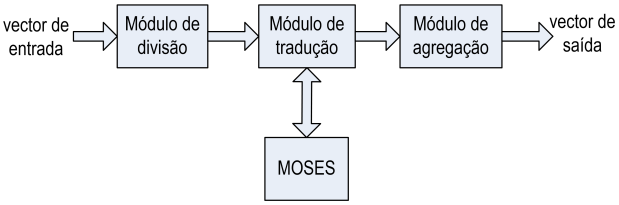


Figura 2: Arquitectura proposta

A parte central deste sistema, e objecto deste artigo, é um módulo – TRADUÇÃO – capaz de criar uma frase, em resposta a um vector com dados, fornecido como entrada. Se o vector fornecido como entrada for demasiado grande, será inicialmente dividido em diversos vectores. Posteriormente, um módulo de agregação irá juntar as diversas frases geradas, produzindo um texto coerente. Para se alcançar este objectivo são necessários três componentes.

O módulo base de dados (BD) é o componente responsável por armazenar todos os tipos de dados, desde as características dos utilizadores, características dos medicamentos, receitas médicas, etc..

O módulo MOSES é responsável pela tradução das frases, enviadas pelo módulo de TRADUÇÃO, para português. Para efectuar este serviço, o Moses precisa de efectuar o seu treino com um corpus, constituído por duas linguagens, perfeitamente alinhadas. A cada frase na linguagem de 'entrada' deve corresponder uma frase na linguagem de 'saída', respeitando o ordenamento dos ficheiros. A linguagem de 'entrada' será constituída por valores correspondentes aos fornecidos pelo módulo de BD. Na linguagem de 'saída' estão as expressões, em português, que se deseja que o Moses seja capaz de gerar.

O módulo de TRADUÇÃO é o módulo principal deste sistema. É responsável por receber os pedidos dos utilizadores e interagir com os dados armazenados na BD, guardando-os ou solicitando-os. É, também, sua responsabilidade enviar mensagens escritas na linguagem de 'entrada' para o módulo MOSES e receber a resposta na linguagem de 'saída'. Por último, compete-lhe processar as respostas e apresentá-las ao utilizador.

3.1 Dois tipos de tradução

O sistema MOSES suporta dois tipos de tradução muito diferentes, conhecidos pelas designações inglesas de *phrase-based* e *tree-based*. Adoptaremos neste artigo as designações de tradução baseada em *sintagmas* e tradução usando *sintaxe*.

Tradução baseada em sintagmas – As denominadas tabelas de tradução são a principal fonte de conhecimento para o *decoder*, que consulta estas tabelas para descobrir como traduzir uma entrada numa linguagem para uma outra linguagem, a de saída.

Estas tabelas de tradução não contêm apenas entradas correspondentes a uma palavra isolada, mas, em geral, as entradas são constituídas por múltiplas palavras. Deste facto deriva a designação de "baseada em sintagmas (frase)". No entanto, neste contexto 'frase' apenas significa uma sequência arbitrária de palavras.

Um exemplo possível de uma entrada na tabela, de acordo com o nosso cenário, seria:

```
Forma2 Medicamento1 ||| comprimidos de
MEDINX ||| 0.8 ||| |||
```

O processo de tradução consiste, como ilustrado na Figura 3, em dividir o vector de entrada em blocos para os quais exista uma tradução e, depois, combinar a saída de cada um desses blocos, com possível reordenação da posição.

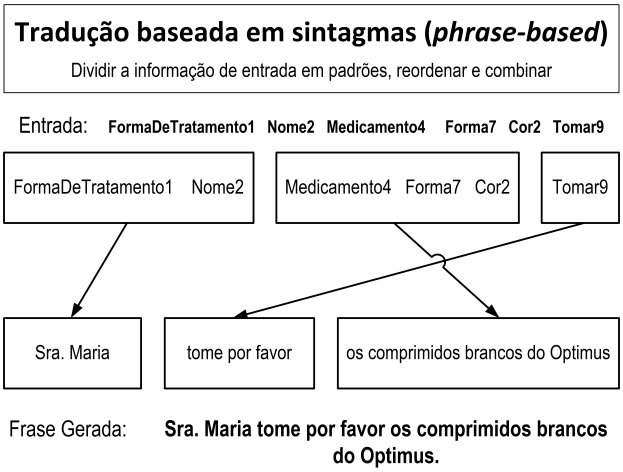


Figura 3: Esquema do funcionamento da tradução baseada em sintagmas

Tradução usando informação sintáctica – Estes modelos são também conhecidos como sistemas hierárquicos baseados em sintagmas e utilizam uma gramática do tipo SCFG (Synchronous Context-Free Grammar). Por esse motivo, quando a 'linguagem de saída' do corpus é processada, cada termo é classificado como 'não-terminal', associando a estes uma etiqueta que representa o seu conteúdo (Nome, Determinante, etc.). Por exemplo, nestes modelos sintácticos, os não-terminais são anotados com etiquetas com informação linguística, como verbo ('VERB') e nome ('NOUN'):

```
NOUN --> tipo1 ||| Comprimidos
VERB --> tomar2 ||| toma
```

Estas etiquetas são obtidas através da utilização de um *parser*, sendo depois utilizadas para efectuar o processo de alinhamento, entre as frases das linguagens de entrada e de saída.

Numa das variantes, que é a que interessa para o presente trabalho, as frases da linguagem de entrada permanecem inalteradas e apenas as de saída sofrem esta transformação. Esta variante é usualmente denominada de frase-para-árvore (string-to-tree, em inglês). As várias permutações – árvore-para-frase e árvore-para-árvore – são também possíveis.

Informação detalhada sobre estes dois tipos de sistemas, assim como sobre a forma de os construir, encontra-se disponível em dois tutoriais mantidos pelos responsáveis pelo sistema MOSES (MOSES, 2014b,c)

Treinados os sistemas, ficam disponíveis os modelos (de linguagem, de tradução, de reordenação) e parâmetros de configuração para a invocação do sistema para processamento de frases na linguagem de entrada. Embora para os dois tipos de sistemas atrás referidos exista um *decoder* MOSES específico, o processo de tradução é conceptualmente similar.

4 Corpus

4.1 Estrutura do corpus

Como referido anteriormente, foi utilizado um corpus constituído por duas linguagens, perfeitamente alinhadas. Por simplicidade, as duas linguagens utilizadas foram designadas por 'linguagem de entrada' e 'linguagem de saída'.

A 'linguagem de entrada' reflecte os dados que são obtidos por consulta, na base de dados, tendo sido seleccionados 9 tipos de dados: Nome e Apelido do utilizador, Mensagem de cortesia, Nome do medicamento a tomar, Tipo do medicamento, Forma de tomar o medicamento, Cor do medicamento, Dose a tomar e Frequência da toma. A estes termos foram concatenados os valores das chaves primárias, correspondentes aos registos seleccionados da base de dados. Desta forma, cada

frase reflecte os dados do utilizador e o que ele deve fazer.

Correspondentemente, na 'linguagem de saída' surge uma frase que exprime o mesmo tipo de informação da 'linguagem de entrada', mas em português. A Tabela 3 apresenta um exemplo destas duas linguagens.

4.2 Obtenção do corpus

O corpus utilizado nesta experiência foi obtido através do seguinte processo:

1. Foi solicitado a um conjunto de 15 voluntários, compreendidos entre os 18 e os 55 anos, que preenchessem um formulário para a introdução de frases, referentes à 'acção de informar uma pessoa sobre os medicamentos que deveria tomar'. Obtiveram-se 126 frases;
2. Aplicando uma estratégia semelhante à descrita em (Langner, 2010), o corpus original foi expandido para um total de 643 frases;
3. Esta expansão foi executada, porque se constatou que cada uma das frases, aplicadas a uma pessoa e medicamento concretos poderia, também, ser aplicada a outras pessoas ou medicamentos, desde que devidamente ajustadas ao novo contexto. Assim, foram executados os seguintes passos:
   (a) Mantendo a sequência original de introdução das frases, a cada frase foram acrescentados, manualmente, *tokens* que identificaram os termos que poderiam ser substituídos. A Tabela 4 apresenta um exemplo de duas frases recolhidas no corpus original e correspondente adaptação com os *tokens*. Na Tabela 5, são apresentados os diversos *tokens* utilizados.
   (b) Após esta fase, cada uma das 126 frases originais do corpus foi replicada entre 3 e 7 vezes, sendo que a geração do número de replicações foi efectuada de forma aleatória. Durante esta replicação, não foi alterada a sequência original do corpus.
   (c) Para permitir a substituição dos *tokens* por nomes de pessoas, medicamentos, etc. foi inicialmente criada uma pequena base de dados com 80 nomes próprios, 33 apelidos e 28 medicamentos. Para cada medicamento, foram identificados o seu nome, o tipo (comprimido, gotas, etc.) e a cor, quando aplicável. Foram, igualmente, identificados 11 períodos possíveis de 'toma' de medicamentos e 6 quantidades diferentes de medicamentos, para cada 'toma'.
   (d) Estas novas 643 frases correspondem à linguagem final do corpus. Ao mesmo tempo, e pela mesma sequência em que se substituía os *tokens* por valores concretos, foi criada uma nova lista de 643 frases, que correspondem à linguagem inicial. O objectivo foi gerar dois ficheiros, cujo conteúdo estivesse perfeitamente alinhado.
   (e) Para cada *token*, em cada frase, foi atribuído um valor escolhido aleatoriamente dentro da base de dados referida. Quando os *tokens* estavam relacionados (MEDICAMENTO, TIPO e COR, por ex.), a escolha de um medicamento implicou a escolha automática para os outros *tokens*, para garantir a integridade da informação.
   (f) A fase final consistiu numa análise, por uma pessoa, das frases obtidas através deste processo de expansão. Foram apenas corrigidos erros gramaticais.

4.3 Preparação do corpus para treino e teste

Usando a técnica *10-fold cross-validation*, descrita, por exemplo, em (Hall et al., 2011; Kohavi, 1995; Salzberg & Fayyad, 1997), o passo seguinte foi a separação do corpus em dois conjuntos disjuntos. Um de teste, com 10% das frases, e outro de treino, com os restantes 90%. Obedecendo a esta métrica, foram gerados 10 grupos distintos. Como as diversas frases da linguagem de saída do corpus foram obtidas, inicialmente, por replicação, se se limitasse a fazer uma separação por simples sorteio aleatório, corria-se o risco de, num mesmo grupo, existir uma maior preponderância de frases com a mesma 'semente'. Por este motivo, as frases foram separadas nos 10 grupos da seguinte forma:

1. Mantendo a sequência inicial das frases, a cada frase foi atribuída uma referência, correspondente a uma letra do alfabeto. Foram utilizadas as letras A a J, identificando cada letra um grupo.
2. A sequência A a J foi atribuída sequencialmente, renovando-se continuamente. Desta forma foi garantido que as frases obtidas a partir de uma dada 'semente', ficam distribuídas por grupos totalmente distintos.

Linguagem interna	**Frase correspondente**
pessoa32n saudacao_0 pessoa0a medicamento21 tipo0 tomar0 cor00 dose0 freqtoma00	Helena pode tomar agora o Seretaide.
pessoa0n saudacao_0 pessoa0a medicamento14 tipo1 tomar2 cor00 dose4 freqtoma02	Vai-se deitar então tome quatro comprimidos Primperan.
pessoa40n saudacao_0 pessoa0a medicamento0 tipo1 tomar2 cor03 dose0 freqtoma04	Ao almoço toma o comprimido branco Leonardo.
pessoa0n saudacao_m pessoa12a medicamento0 tipo8 tomar3 cor00 dose0 freqtoma02	Antes de deitar senhor Lima não se esqueça da bomba de inalação.
pessoa17n saudacao_f pessoa0a medicamento0 tipo4 tomar2 cor04 dose0 freqtoma02	Antes de deitar faça a toma das gotas amarelas dona Cristina.
pessoa0n saudacao_0 pessoa0a medicamento3 tipo1 tomar2 cor10 dose4 freqtoma05	É hora de jantar tome os quatro comprimidos laranja do Ibuprofeno.
pessoa36n saudacao_0 pessoa0a medicamento19 tipo8 tomar3 cor00 dose0 freqtoma01	São horas de acordar e de colocar a bomba de inalação Nasomet daqui a três horas João terá de colocar de novo.
pessoa21n saudacao_f pessoa0a medicamento2 tipo4 tomar2 cor00 dose5 freqtoma01	Dona Elisabete assim que acordar deve tomar cinco gotas de Clorocil.
pessoa37n saudacao_0 pessoa0a medicamento23 tipo4 tomar2 cor00 dose4 freqtoma05	Está na hora de jantar Jorge não esqueça de tomar as quatro gotas de Guttalax.
pessoa78n saudacao_f pessoa0a medicamento3 tipo1 tomar2 cor00 dose3 freqtoma04	Dona Teresinha está na hora de almoço tome os três comprimidos Ibuprofeno.

Tabela 3: Parte do corpus alinhado utilizado nas experiências (as 10 primeiras linhas do corpus de treino A).

Adriana podes tomar agora o Clorocil
D. Teresa antes de deitar coloque as gotas Clorocil
NOME podes tomar agora o MEDICAMENTO
FORMA_TRATAMENTO NOME antes de deitar coloque as TIPO MEDICAMENTO

Tabela 4: Duas frases recolhidas no corpus original e correspondente adaptação com os *tokens*.

Token	**Correspondência**
NOME	nome do destinatário do medicamento
APELIDO	apelido do destinatário do medicamento
FORMA_TRATAMENTO	saudação ao destinatário do medicamento (corresponde a Sr., Sra., D., etc.)
MEDICAMENTO	nome do medicamento
QUANTIDADE	quantidade a tomar do medicamento
TIPO	tipo de medicamento (comprimidos, gotas, etc.)
TEMPO	hora do dia em que o medicamento deveria ser tomado
COR	cor do medicamento

Tabela 5: *Tokens* e respectivas correspondências.

3. Após a classificação das frases, o corpus foi ordenado pela sua referência (letra A a J), constituindo-se assim 10 grupos disjuntos.

4. Ao fazer a separação do corpus desta forma foi garantida a aleatoriedade da constituição de cada grupo. A produção inicial de cada uma das frases 'semente' é independente. A replicação das 126 frases para as 643 finais é aleatória, já que cada frase foi replicada, aleatoriamente, entre 3 e 7 vezes. A substituição dos *tokens* por valores concretos foi efectuada com valores escolhidos aleatoriamente. Na distribuição de frases por grupos, todas as frases de cada grupo têm uma 'semente' distinta.

Concretizada a separação do corpus em 10 grupos disjuntos, foram constituídos 10 conjuntos de teste e de treino. Cada conjunto, igualmente denominado por uma letra de A a J e sufixado respectivamente por "-teste" e "-treino", corresponde ao seguinte: o conjunto de teste tem o nome igual ao do grupo obtido, como acima explicado; O conjunto de treino contém as frases de todos os restantes grupos.

4.4 Algumas estatísticas

Alguma informação estatística acerca da 'linguagem de saída' do corpus pode ser encontrada na Tabela 6.

Número de frases	643
Número de palavras	7212
Número médio de palavras/frase	11
Número máximo de palavras/frase	30
Número mínimo de palavras/frase	4

Tabela 6: Alguma informação estatística relativa à 'linguagem de saída' do corpus.

5 Sistemas Desenvolvidos

Foram desenvolvidas duas variantes do sistema, aproveitando as duas variantes principais dos sistemas de tradução automática: baseadas em sintagmas (*phrase-based*) e usando informação sintáctica (*tree-based*).

5.1 Sistemas baseados em sintagmas (*phrase-based*)

Para a execução desta primeira experiência, após a recolha do corpus, respectiva expansão e segmentação procedeu-se à operação de treino. Foram executados os diversos procedimentos, conforme prescrito em (MOSES, 2014a). O Moses dispõe de diversos *scripts* especialmente preparados para a execução de cada uma das fases. Para cada um dos 10 conjuntos de treino, foi efectuada uma operação de treino, e teste, totalmente independente. Cada conjunto foi submetido aos mesmos procedimentos, executados pela mesma ordem.

A primeira tarefa consistiu na preparação do corpus. Em primeiro lugar foi efectuada a *tokenização* do corpus. Esta operação consiste em separar, com um espaço em branco, antes e depois, cada um dos elementos que constituem cada uma das frases do corpus. Aqui, por "elemento" entende-se cada palavra e sinal de pontuação existentes na frase. Na 'linguagem de saída' este procedimento foi realizado com recurso ao script 'tokenizer.perl'. Na 'linguagem de entrada' não foi necessário efectuar esta tarefa, pois da forma como ela foi criada, todos os elementos de cada frase estão naturalmente *tokenizados*.

A fase de limpeza foi executada, contudo não teve efeitos práticos. Efectivamente, esta fase destina-se a eliminar as frases mal formadas e com tamanho excessivo. Considera-se tamanho excessivo uma frase com mais de 80 palavras. No nosso corpus, todas as frases têm dimensão inferior a esse limite e encontram-se bem formadas e devidamente alinhadas.

A segunda tarefa consistiu no treino do modelo de linguagem. Nesta fase, são criadas as ferramentas intermédias que vão assistir o Moses na realização do treino do sistema de tradução. Este modelo intermédio destina-se a assegurar uma geração fluente do texto produzido, sendo por isso efectuada sobre a 'linguagem de saída'. Neste passo, foi utilizado o IRSTLM (IRSTLM, 2011).

Neste mesmo passo, é, também, definido o parâmetro *ngram*. Por defeito, tem o valor 3, o que significa que o modelo irá efectuar agrupamentos de palavras até 3 elementos. Estes agrupamentos serão posteriormente utilizados na criação dos textos de saída. São essencialmente executados 3 passos. Primeiro, cada frase é prefixada com o termo <s>e sufixada com o correspondente termo </s>. Depois, o modelo da linguagem é construído. Por último, este modelo é compilado.

A terceira tarefa consistiu no treino do sistema de tradução. Concluídas as duas primeiras tarefas, estão reunidas as condições para se efectuar o treino do sistema de tradução. Esta tarefa recorre ao modelo da linguagem, gerado na tarefa anterior, e ao software GIZA++ (Och, 2011). Aqui são gerados, entre outros, os ficheiros 'moses.ini' e 'phase-table.gz' necessários à configuração e utilização do Moses.

A última tarefa consiste no teste, e uso, do modelo treinado para se gerarem os textos pretendidos.

5.2 Sistemas usando informação sintáctica (*tree-based)*

Na execução com o modo *tree-based*, foram utilizados os mesmos conjuntos, já referidos, tendo sido realizadas experiências independentes. A grande diferença entre o treino anterior e este treino centra-se na construção da árvore que representa a 'linguagem de saída'. Posteriormente,

é o ficheiro com a árvore que é utilizado no treino do sistema de tradução.

A produção da árvore, que permitiu classificar cada palavra, de cada frase, em termos da sua função morfossintáctica (nome, verbo, adjectivo, etc.), revelou-se uma tarefa difícil de superar.

A principal dificuldade resultou da escolha do *parser* a utilizar. O nossos principais requisitos eram: (1) produzir uma classificação que pudesse ser facilmente adaptada para utilização no MOSES e que fosse compatível com as ferramentas de manipulação de "árvores" deste sistema; (2) possibilidade de integração do *parser* no nosso sistema; (3) utilização gratuita.

Consideradas estas exigências, foram instalados e testados diversos *parsers*, nomeadamente: o Palavras[1], o Freeling[2], o Tycho Brahe[3], o TreeTagger[4], o Turbo Semantic Parser[5] e o Stanford Parser[6].

A escolha recaiu sobre este último, com a adaptação efectuada pelo grupo LX-CENTER (Language Resources and Technology for Portuguese), da Universidade de Lisboa – Portugal (Branco & Silva, 2004). Apesar das limitações verificadas, foi, no entanto, o *parser* que melhor correspondeu aos nossos requisitos.

6 Resultados

Nesta secção, apresentam-se exemplos representativos das capacidades de geração dos sistemas desenvolvidos. Segue-se a apresentação de uma avaliação formal, usando métricas comuns na área e avaliação por humanos. A eventual influência da forma de divisão do corpus (em treino e teste) é também avaliada. No final, apresenta-se alguma informação sobre a integração em curso na aplicação para *Smartphones* denominada Assistente de Medicação.

6.1 Exemplos

Na Tabela 7, apresentam-se vários exemplos seleccionados de forma a ilustrar os vários tipos de resultados obtidos. Pretende-se familiarizar o leitor com o que de facto foi possível obter, usando os dois tipos de sistemas.

No topo da tabela, com os números 1 a 3, apresentam-se alguns exemplos de representação interna e as correspondentes frases gerada por um dos sistemas baseado em sintagmas (concretamente o treinado com a parte A do corpus). As frases geradas têm qualidade bastante diversa, sendo uma intelegível, outra considerada aceitável e a restante considerada correta.

Na segunda parte da tabela (números 4 a 7) apresenta-se o alinhamento entre as frases criadas pelos humanos e as criadas pelo sistema (independentemente do seu tipo) para uma mesma entrada. As frases são, aqui, apresentadas em minúsculas para que seja possível evidenciar as suas diferenças, como a seguir explicado. Para uma certa frase de entrada, quando uma frase é gerada pelo nosso sistema, ela pode apresentar (e normalmente apresenta) diferenças com a frase de treino. Essas diferenças podem corresponder a adições de novas palavras, supressão de palavras ou troca de posição de palavras dentro da frase. Aqui, utilizamos a marcação "***" para evidenciar a adição ou supressão de palavras e as maiúsculas para assinalar a troca de posição de palavras. Estas ocorrências surgem apenas na frase gerada. Quando as representamos na frase escrita por humanos, pretendemos, apenas, tornar mais evidente as diferenças entre as duas frases.

Os exemplos 6 e 7 apresentam frases geradas, consideradas inteligíveis e boas em termos de naturalidade, mas que são completamente diferentes das produzidas por humanos. Este tipo de frases constitui um grande desafio para a avaliação, sendo normalmente consideradas como erros pelas métricas automáticas de avaliação. Tendo em conta estas ocorrências, tivemos necessidade de reconsiderar a nossa opção inicial de apenas aplicar avaliação automática, avaliando uma parte dos resultados por humanos.

Na terceira parte da tabela, exemplifica-se a diferença entre os resultados obtidos pelos 2 tipos de sistema. Para cada um dos subconjuntos, foi utilizado o mesmo vector de entrada, para garantir a comparabilidade das frases.

6.2 Avaliação comparativa dos dois tipos de sistemas

6.2.1 Método

Tendo em conta os objectivos principais de ter informação sobre o desempenho absoluto e relativo dos dois tipos de sistemas criados, começou-se por treinar 10 sistemas para cada um dos tipos, adoptando-se os valores por defeito para a generalidade dos processos (ou seja o valor de 3 para o

[1] http://beta.visl.sdu.dk/contact.html
[2] http://nlp.lsi.upc.edu/freeling/
[3] http://www.tycho.iel.unicamp.br/~tycho/apps/dbparser-files/
[4] http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/DecisionTreeTagger.html
[5] http://labs.priberam.com/Resources/TurboSemanticParser.aspx
[6] http://nlp.stanford.edu/software/lex-parser.shtml

Exemplos mostrando a linguagem de entrada e a frase gerada	
Num	**Exemplo**
1	pessoa45n saudacao0 pessoa0a medicamento17 tipo0 tomar0 cor00 dose0 freqtoma00 Luís Pulmicort de tomar agora o
2	pessoa61n saudacaom pessoa0a medicamento4 tipo0 tomar0 cor00 dose2 freqtoma00 senhor Paulo tome dois comprimidos de Salazopirina
3	pessoa49n saudacao0 pessoa0a medicamento0 tipo2 tomar2 cor09 dose0 freqtoma10 Marcelo aplique ao tomar a cápsula branca e azul de dez horas em dez horas
Exemplos de saída dos sistemas (S) alinhados com frases produzidas por humanos (H)	
4	H: dona denise assim que se levantar não se esqueça de tomar OS COMPRIMIDOS nicotibine S: dona denise assim que se levantar não se esqueça de tomar O COMPRIMIDO nicotibine
5	H: DEVE TOMAR AGORA ao acordar *** a bomba de inalação DE pulmicort AUGUSTO S: *** *** AUGUSTO ao acordar APLIQUE a bomba de inalação *** pulmicort ***
Exemplos de geração muito diferentes da frase de teste, mas aceitáveis	
6	H: É HORA DE ALMOÇAR marcos não se esqueça de tomar *** quatro gotas de guttalax *** *** S: *** *** *** *** marcos não se esqueça de tomar AS quatro gotas de guttalax AO ALMOÇO
7	H: *** *** É MEIO-DIA TOME AS três gotas de zaditen *** PATRÍCIA S: PATRÍCIA NÃO SE ESQUEÇA DE TOMAR três gotas de zaditen AO MEIO-DIA
Saídas produzidas pelos 2 sistemas, para uma mesma entrada (F identifica o baseado em sintagmas, S2T identifica o baseado em informação sintáctica)	
8	F: Patrícia não se esqueça de tomar três gotas de Zaditen ao meio-dia S2T: Patrícia ao Zaditen gotas tome de três meio-dia
9	F: Senhora Carvalho após o seu almoço tome cinco comprimidos de Duphaston S2T: Senhora Carvalho Duphaston comprimido branco cinco almoço

Tabela 7: Exemplos de saídas dos dois tipos de sistemas.

ngram). Depois de treinados, os sistemas foram avaliados, primeiro com o corpus de treino e, após verificado o bom funcionamento do sistema, com o conjunto de teste correspondente. Adoptaram-se para a avaliação, os parâmetros BLEU (Papineni et al., 2002) e Meteor (Denkowski & Lavie, 2014, 2011), seguindo, por exemplo (Langner, 2010).

Complementarmente, foi realizada uma avaliação por humanos. As frases avaliadas foram escolhidas de entre as geradas pelos 2 sistemas. Todas estas frases foram obtidas com o mesmo conjunto de teste — o conjunto F. Depois de escolhidas, foram ordenadas aleatoriamente e avaliadas em termos de inteligibilidade, estrutura da frase e qualidade global. Para simplificar a tarefa dos avaliadores, as respostas possíveis para inteligibilidade e estrutura foram reduzidas a apenas 3 opções. Informação concreta sobre as questões e as opções de resposta encontram-se na Tabela 8. Participaram na avaliação 11 pessoas, com características muito diferentes, quer em termos de idade (variaram entre os 16 anos e os 58 anos), quer em termos de formação e actividade profissional (e.g. estudantes, assistentes administrativos e professores).

	Questão	**Opções de resposta**
Inteligib.	Percebe-se ?	0 = Não 1 = Mais ou menos 2 = Sim
Estrutura	Estrutura da Frase	0 = Má (vários problemas) 1 = Mais ou menos 2 = Boa
Qualidade	Qualidade Geral ?	de 1 a 5, onde 1 = Má 5 = Excelente

Tabela 8: Informação sobre as questões e opções de resposta utilizadas na avaliação das frases geradas por humanos.

No caso dos sistemas baseados em sintaxe, foi feita uma experiência relativa ao efeito do peso atribuído ao modelo de linguagem no processo de *decoding*. Foram experimentados vários pesos, usando o corpus de treino, tendo-se chegado à conclusão de que existia um efeito muito positivo nas métricas BLEU e Meteor quando esse peso era bastante superior ao valor por omissão. Tendo em conta este resultado, este novo valor (de 10) foi adoptado para as avaliações de todos os sistemas baseados em sintaxe.

6.2.2 Resultados da avaliação automática

Os resultados obtidos em termos de BLEU e Meteor para os dois tipos de sistemas, são apresentados nas Figuras 4 e 5, sob a forma de média e intervalo de confiança a 95%.

Em termos de BLEU, na Figura 4, o sistema baseado em sintagmas obtém um melhor desempenho médio, em todos os parâmetros, excepto para o parâmetro Bleu 1 – associado a unigramas –, sendo mesmo o desempenho significativamente superior (visível pela não sobreposição dos intervalos de confiança) para o parâmetro global (BLEU na figura) e para o Bleu 2. O melhor valor de BLEU obtido foi de 0,245.

Os resultados anteriores são, de um modo geral, confirmados pelo Meteor (ver Figura 5). Neste caso, todos os parâmetros são piores para o sistema baseado em sintaxe, sendo de destacar as diferenças em termos de 'Recall', 'Penalização Devida a Fragmentação' e na 'Score Final'. Enquanto ambos os sistemas são capazes de um desempenho similar em termos de precisão, acertando em geral nas palavras que seleccionam para a frase, o sistema baseado em sintaxe falha muito mais na inclusão de palavras que deviam constituir a frase.

Os resultados obtidos pelo sistema baseado em sintaxe estão certamente relacionados com a qualidade da classificação sintáctica efectuada pelo *parser* escolhido. A Figura 6 apresenta três exemplos de frases com problemas na classificação. São evidentes erros na classificação de verbos, nomes e artigos. O problema mais recorrente é a deficiente classificação dos nomes dos medicamentos.

```
(ROOT (S
   (NP (N Daniel))
   (VP (V deve) (VP (V' (V tomar) (ADV agora))
   (NP (NP (ART o) (N comprimido))
   (S (VP (V Duphaston) (NP (ART uma) (N' (N vez) (CP (NP (REL que))
   (S (VP (V são) (NP (CARD vinte) (N horas)))))))))))))

(ROOT (S
   (NP (NP (N' (N No) (N fim) (N do) (N jantar)))
   (NP (N' (N dona) (N Inês))))
   (VP (V aplique) (NP (ART a) (N' (N pomada) (A Fucithalmic))))))

(ROOT (S
   (S (CONJ Quando) (S (NP (CL se)) (VP (V levantar)
      (NP (N' (N senhora)
      (N Pereira) (N tome))))))
   (S (NP (ART as) (N' (CARD duas) (N gotas)))
      (VP (V' (V Neo-Sinedrina)
      (ADV depois))
      (NP (N' (N repita) (N ao) (N almoço)))))))
```

Figura 6: Exemplos de erros na classificação de palavras, originado pelo *parser* Stanford.

A pequena extensão dos intervalos de confiança mostra uma pequena variação dos resultados com a divisão do corpus utilizada para teste.

Da conjugação dos resultados obtidos nas duas métricas de avaliação, resulta claro um melhor desempenho do sistema baseado em sintagmas para esta tarefa.

6.2.3 Resultados da avaliação por humanos

Os resultados da avaliação por humanos são resumidos nas Figuras 7 a 9. Em cada figura, são apresentadas as contagens de cada uma das opções de resposta, em confronto com os resultados dos dois tipos de sistema.

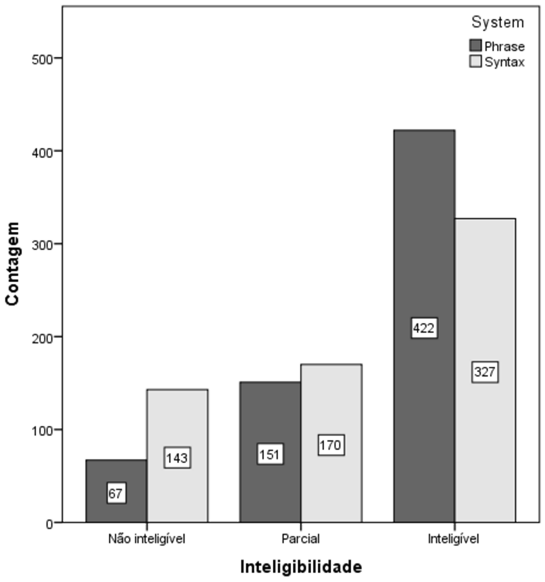


Figura 7: Distribuição das respostas da questão relativa à inteligibilidade das frases. As barras mais escuras referem-se ao sistema baseado em sintagmas.

Em termos de inteligibilidade, o sistema baseado em sintagmas obteve um maior número de respostas, indicando frases inteligíveis (uma diferença de 95 respostas, o que significa uma média de mais 8,7 respostas positivas em termos de inteligibilidade por avaliador). Sendo a diferença entre os dois sistemas para avaliações indicando inteligibilidade parcial baixa, este pior desempenho do sistema baseado em sintaxe reflecte-se no maior número de frases avaliadas como não inteligíveis. Enquanto o sistema baseado em sintagmas apresenta uma média de 6 frases não inteligíveis por avaliador, o sistema baseado em sintaxe apresenta um valor médio superior ao dobro (13). Tendo em conta que foram avaliadas 64 frases para cada sistema, estes valores correspondem a, respectivamente, 9 e 20 % de frases não-inteligíveis. Do lado positivo, foram consideras como inteligíveis, por cada avaliador, uma média de 60 % e 46 %.

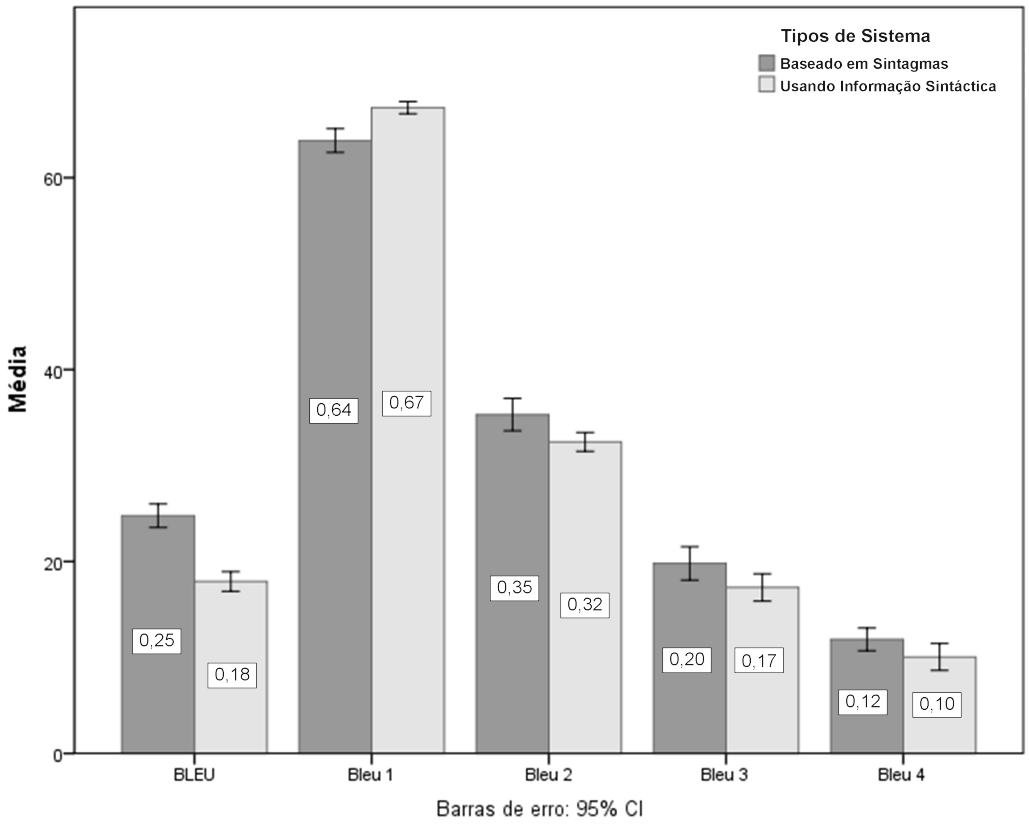


Figura 4: Resultados da avaliação baseada no BLEU.

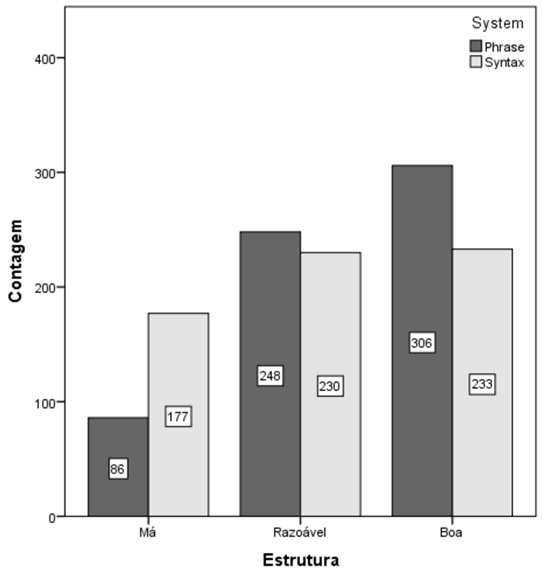


Figura 8: Resultados da avaliação da estrutura das frases. As barras mais escuras correspondem ao sistema baseado em sintagmas.

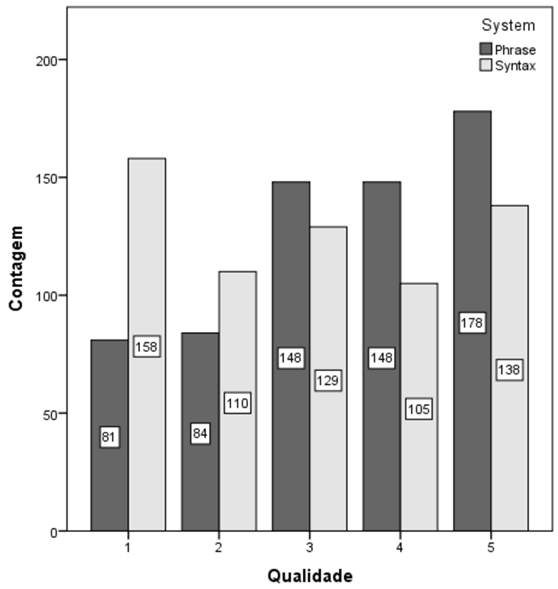


Figura 9: Distribuição das respostas da questão relativa à qualidade geral das frases. As barras mais escuras referem-se ao sistema baseado em sintagmas.

Em termos de estrutura das frases (Figura 8), mantém-se o melhor desempenho do sistema baseado em sintagmas, com um valor próximo do dobro de frases avaliadas como tendo vários problemas de estrutura. Mais uma vez, as grandes diferenças ocorrem nos extremos (má estrutura e boa estrutura).

Na avaliação geral da qualidade (Figura 9), o sistema baseado em sintagmas apresenta número de avaliações superiores ao baseado em sintaxe para os valores geralmente interpretados como indicando uma avaliação positiva (iguais ou su-

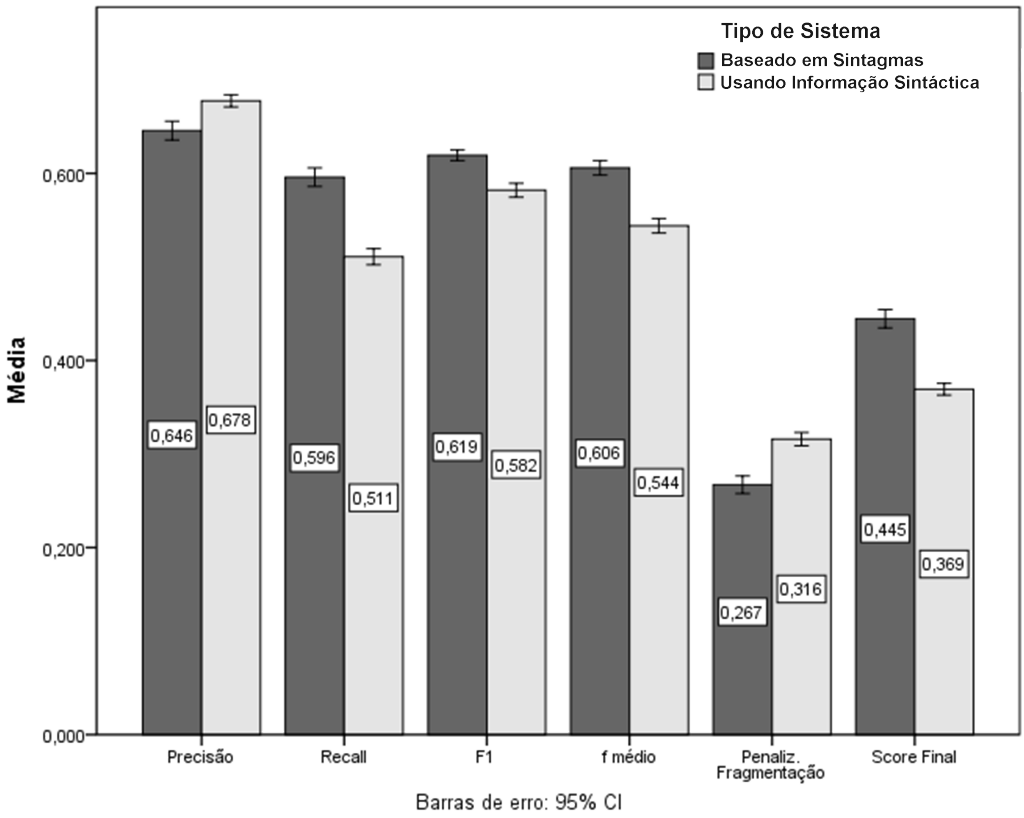


Figura 5: Resultados da avaliação usando o Meteor.

periores a 3). O sistema baseado em sintaxe obteve um valor muito superior de avaliações com o valor mais baixo da escala (1). Ambos os sistemas apresentam uma distribuição das classificações pelos 5 valores da escala, com tendência, no caso do baseado em sintagmas, para uma preferência pelos valores entre 3 e 5. É importante destacar que, para o melhor sistema, foram, em média, avaliadas como boas ou excelentes 46 % e como excelentes 25 % das frases.

6.3 Efeito da forma de divisão do corpus

De forma a descartar um possível efeito do modo como foi dividido inicialmente o corpus (ver Secção 4.3), foi criada uma nova experiência, onde o corpus foi dividido aleatoriamente em 10 novos conjuntos, com número similar de frases. Nesta experiência, tendo em consideração os melhores resultados inicialmente obtidos, foi decidido utilizar a versão baseada em sintagmas. Estes novos conjuntos foram treinados e testados de forma equivalente à experiência anterior. A Figura 10 evidencia as diferenças obtidas nas duas experiências.

É notório para os dois conjuntos de métricas que o desempenho, avaliado nos 10 conjuntos de teste, é superior quando utilizado o método de divisão proposto neste artigo. Esta diferença é estatisticamente significativa para um nível de confiança de 5 % (os intervalos de confiança, CI, a 95 % não se sobrepõem).

Tendo em conta os resultados superiores com a divisão não-aleatória do corpus, decidiu-se pela não realização de mais treinos e testes com as divisões obtidas pelo processo de divisão aleatória.

6.4 Variabilidade das respostas do sistema

Um bom sistema baseado em *templates* pode produzir interacção com o ser humano com muita naturalidade, variabilidade e qualidade. Contudo, para que tal aconteça, é necessário efectuar um grande investimento, seja em tempo, seja em recursos.

Experiências efectuadas com o nosso sistema mostraram que, para um vector de entrada similar, são produzidas respostas corretas e distintas, que não fazem parte do corpus inicial. Este tipo de geração permite criar, com facilidade, novas respostas, proporcionando variabilidade na interacção com o seu utilizador. A Tabela 9 apresenta dois exemplos onde, para entradas similares, obtemos respostas distintas. Nos exemplos 1 e 2, apenas varia o nome da pessoa referenciada. Nos exemplos 3, 4 e 5, apenas varia o nome do medicamento, e naturalmente, a sua forma e tipo de toma.

Num	Exemplo
1	pessoa18n saudacao_m pessoa0a medicamento24 tipo3 tomar1 cor00 dose0 freqtoma20 Senhor Daniel aplique a pomada Fucithalmic que são vinte horas
2	pessoa18n saudacao_m pessoa5a medicamento24 tipo3 tomar1 cor00 dose0 freqtoma20 Senhor Daniel Costa deve aplicar a pomada Fucithalmic que são vinte horas
3	pessoa18n saudacao_m pessoa0a medicamento24 tipo3 tomar1 cor00 dose0 freqtoma20 Senhor Daniel aplique a pomada Fucithalmic que são vinte horas
4	pessoa18n saudacao_m pessoa0a medicamento3 tipo1 tomar2 cor00 dose0 freqtoma20 Senhor Daniel não se esqueça de tomar o comprimido Ibuprofeno são vinte horas
5	pessoa18n saudacao_m pessoa0a medicamento12 tipo1 tomar2 cor00 dose0 freqtoma20 Senhor Daniel tome o comprimido de Nicotibine são vinte horas

Tabela 9: Exemplo de geração de respostas, associadas a entradas similares.

6.5 Primeiras integrações

Uma versão inicial dos sistemas aqui apresentados, baseada em sintagmas, foi alvo de uma primeira integração numa aplicação real para assistência à toma de medicamentos por idosos. Este sistema, para Smartphones, e desenvolvido no âmbito do projecto *Smartphones for Seniors*, foi descrito e avaliado em (Teixeira et al., 2013a; Ferreira et al., 2013a,b)

7 Discussão

Os resultados obtidos pelo melhor sistema, em termos de inteligibilidade e qualidade das frases geradas, indicam que a abordagem adoptada e os sistemas desenvolvidos conseguem gerar frases de qualidade similar às produzidas por humanos. Contudo quando falham, as frases geradas podem ser completamente ininteligíveis.

Estes resultados estão de acordo com os relatados por Langner (Langner, 2010) para o sistema Mountain. No Mountain, foi apenas utilizada a variante de desenvolvimento por sintagmas (*phrase-based*), pelo que o nosso comentário se refere apenas a este tipo de geração. A Tabela 10 apresenta os resultados por ele obtidos. As variantes ao sistema base (*Rating >1 . . . Rating >4*) correspondem a experiências efectuados por Langner. Essas experiências foram motivadas pela qualidade do corpus utilizado no Mountain. Cada experiência corresponde, assim, a retiradas sucessivas, ao corpus, das frases classificadas, por avaliadores humanos, como inadequadas. Verifica-se que, até certo ponto, quanto menos frases consideradas 'más' estiverem presentes no corpus, melhor o índice BLEU.

Da comparação entre os dados da Figura 4 e da Tabela 10 resulta que o nosso sistema apresenta valores ligeiramente superiores ao Mountain (qualquer que seja a sua versão), quando se comparam os dados referentes aos Bleu1, Bleu2, Bleu3 e Bleu4.

A Tabela 11 apresenta os dados da avaliação do sistema Mountain, nas mesmas condições referidas atrás, segundo o método Meteor. Comparando os dados desta figura com os dados da Figura 5, verifica-se que, à semelhança do observado para o método BLEU, os resultados do nosso sistema são ligeiramente melhores, nos seus diversos indicadores.

Estas observações permitem-nos concluir que, apesar das limitações do nosso sistema, o seu desempenho é satisfatório e está alinhado com o desempenho de sistemas similares.

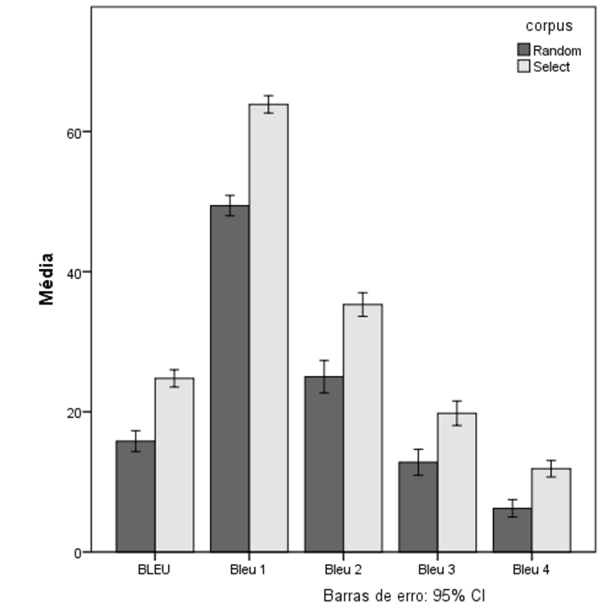
System	1-gram	2-gram	3-gram	4-gram	5-gram
Baseline	0.3198	0.1022	0.0525	0.0300	0.0202
Rating > 1	0.4376	0.1729	0.1079	0.0746	0.0597
Rating > 2	**0.4491**	**0.1919**	0.1169	0.0872	0.0747
Rating > 3	**0.4742**	**0.1963**	0.1212	0.0866	0.0722
Rating > 4	**0.4596**	0.1762	0.1023	0.0693	0.0611

Tabela 10: Avaliação do sistema Mountain (Langner, 2010, pag. 73), pelo método BLEU.

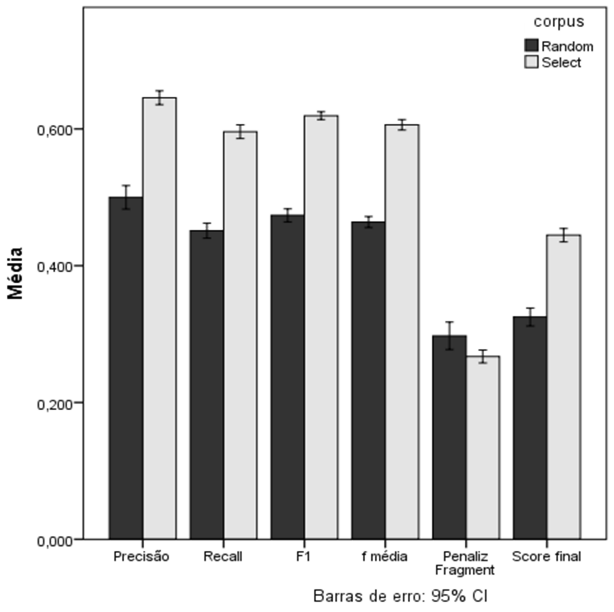
System	Precision	Recall	F_1	Total
Baseline	0.4225	0.2013	0.2727	0.1950
Rating > 1	0.4489	0.2097	0.2859	0.2028
Rating > 2	0.4533	**0.2248**	**0.3009**	**0.2218**
Rating > 3	**0.4834**	0.2148	0.2974	0.2146
Rating > 4	0.4481	0.2030	0.2794	0.1971

Tabela 11: Avaliação do sistema Mountain (Langner, 2010, pag. 75), pelo método Meteor.

Um dos objectivos que pretendemos alcançar, é a geração de um sistema em que o esforço necessário para a obtenção de corpora seja pequeno. Tendo em conta este requisito, os sistemas foram capazes de gerar um conjunto interessante de frases, apesar de terem sido treinados apenas com um pequeno corpus, criado tendo por base pouco mais de 100 frases efectivamente escritas por humanos. Este facto demonstra não só o potencial para melhoramento, através do aumento do corpus, mas também o potencial para se criar sistemas minimamente úteis com muito pouco investi-



a) BLEU



b) METEOR

Figura 10: Resultados obtidos com os dois métodos de divisão do corpus utilizados. Os resultados referem-se apenas aos sistemas baseados em sintagmas.

mento na criação de corpora. O melhor desempenho do sistema mais simples (baseado em sintagmas) não deve ser interpretado como um indício da desadequação do sistema usando informação sintáctica. Como potenciais causas deste desempenho inferior temos: (1) reduzido tamanho do corpus, que pode ser insuficiente para treinar adequadamente os modelos, certamente mais exigentes; (2) efeito negativo dos erros de análise sintáctica. Quanto ao segundo problema, estamos convictos que, com a utilização de um *parser* que classifique melhor as diversas palavras das frases, quanto à sua função sintáctica, o desempenho do sistema baseado em sintaxe melhorará.

Por outro lado, o facto de os dois sistemas gerarem muitas vezes frases muito diferentes pode ser explorado na criação de um sistema em que os resultados de ambos sejam objecto de um processo de avaliação e selecção da melhor frase. Acreditamos que o desempenho de ambos os sistemas é passível de ser melhorado, através de um processo de afinação (*tuning*) dos muitos parâmetros dos modelos, usando um corpus de validação.

Um outro aspecto positivo a destacar é o aumento do desempenho obtido, através de um método não-aleatório de divisão do corpus em treino e teste. uma vez que se aplicou um processo de expansão de um conjunto base de frases na criação do corpus, o método usual de divisão aleatória não é o mais adequado, não garantindo que exemplos derivados de uma mesma frase fiquem devidamente divididos entre os conjuntos de treino e teste. O método proposto evita que se reduza em demasia a presença, no conjunto de treino, de exemplos resultantes de uma mesma frase base.

A grande limitação dos sistemas criados é a sua imprevisibilidade em termos de qualidade dos resultados. A avaliação, como esperado, revelou-se uma tarefa bastante complicada, com as métricas automáticas a apresentar grandes dificuldades em fornecer informação adequada. Apenas com a utilização combinada de 2 métricas e avaliação por humanos foi possível ter uma visão minimamente clara sobre o desempenho dos sistemas. Recentemente, foi apresentada (Pereira et al., 2015) uma primeira extensão a este sistema, onde é feita uma proposta de avaliação automática da qualidade, recorrendo à extração e análise de características (*features*) sobre as frases geradas.

8 Conclusão

Motivados pela crescente necessidade de transmitir, em português, informação gerada por sistemas computacionais cada vez mais sofisticados e omnipresentes, neste artigo apresenta-se uma primeira experiência para a língua portuguesa na área da geração de frases a partir de dados referentes a planos de medicação. O sistema adoptado baseia-se na utilização de tradução automática e inspira-se em trabalhos recentes, como o sistema Mountain. Foram desenvolvidos e avaliados comparativamente dois tipos de sistemas

de tradução, um baseado em sintagmas e outro usando informação sobre a sintaxe. Os resultados da avaliação, englobando avaliação automática e avaliação por humanos, mostram que o sistema baseado em sintagmas obteve o melhor desempenho. Este tipo de sistemas é capaz de gerar uma boa percentagem de frases inteligíveis ou minimamente inteligíveis (menos de 10 % de frases não inteligíveis), com percentagem interessante das frases geradas a obter avaliações de qualidade geral de nível bom ou superior.

Para divisão do corpus para o treino de teste de 10 sistemas (*10-fold Cross-Validation*) foi desenvolvido um processo alternativo à usual divisão aleatória, que se revelou capaz de contribuir para um melhor desempenho dos sistemas testados.

8.1 Trabalho Futuro

Uma continuação óbvia do trabalho aqui apresentado passa pelo aumento do corpus. Para esta primeira experiência considerou-se importante não investir muitos recursos na criação de um corpus extenso, mas é importante investigar o efeito do tamanho do corpus no desempenho deste tipo de sistema.

Não produzindo os sistemas desenvolvidos uma percentagem de frases inteligíveis próxima dos 100 %, de forma a tornar estes sistemas utilizáveis numa aplicação real, como é nosso objectivo, torna-se necessário desenvolver um módulo que disponibilize uma estimativa da inteligibilidade e naturalidade das frases geradas. Com essa informação, será possível criar um sistema híbrido que recorra a *templates* quando essa estimativa aponte para uma frase de baixa qualidade e em que a inteligibilidade esteja comprometida.

Nesta fase do nosso trabalho, a nossa principal preocupação, foi determinar se as frases geradas eram facilmente compreendidas por qualquer pessoa. A etapa seguinte será avaliar a sua adequabilidade, com recurso a profissionais da área do tema da aplicação.

Por último, mas não menos importante, interessa-nos explorar formas rápidas de aplicar este tipo de abordagens a outras aplicações. Interessa-nos, também, reforçar a multimodalidade da aplicação. Nomeadamente, com recurso à síntese de fala e/ou com recurso a imagens. Só desta forma será possível complementar a informação prestada e eliminar algumas ambiguidades que eventualmente existam.

Agradecimentos

Os autores agradecem a todos os que contribuíram para a criação do corpus e a todos os que participaram na avaliação das frases. Um agradecimento especial ao Mário Rodrigues pela ajuda na obtenção e utilização dos *parsers* sintácticos para o português.

Um último agradecimento para os vários revisores deste artigo, que em muito contribuíram para a sua evolução.

Referências

Agência Lusa. 2010. Falantes de português irão aumentar para 335 milhões em 2050. *Público (Online, verificado em 23/07/2015)* http://www.publico.pt/culturaipsilon/noticia/falantes-de-portugues-irao-aumentar-para-335-milhoes-em-2050-1429372.

Alves, Lúcia Vinheiras. 2011. Português é terceira língua mais falada no mundo. Online, verificado em 23/07/2015. http://www.tvciencia.pt/tvcnot/pagnot/tvcnot03.asp?codpub=26&codnot=8.

de Araújo, Roberto P. A., Rafael L. de Oliveira, Elder M. de Novais, Thiago D. Tadeu, Daniel B. Pereira & Ivandré Paraboni. 2010. SINotas: the Evaluation of a NLG Application. Em *Proc. Seventh International Conference on Language Resources and Evaluation*, 2388–2391.

Bateman, John & Michael Zock. 2004. Natural Language Generation. Em Ruslan Mitkov (ed.), *The Oxford Handbook of Computational Linguistics*, 284–304. Oxford University Press.

Branco, António & João Silva. 2004. Evaluating solutions for the rapid development of state-of-the-art POS taggers for portuguese. Em *4th International Conference on Language Resources and Evaluation (LREC)*, 507–510.

Denkowski, Michael & Alon Lavie. 2011. Meteor 1.3: Automatic Metric for Reliable Optimization and Evaluation of Machine Translation Systems. Em *Proceedings of the EMNLP 2011 Workshop on Statistical Machine Translation*, 85–91.

Denkowski, Michael & Alon Lavie. 2014. Meteor universal: Language specific translation evaluation for any target language. Em *Proceedings of the EACL 2014 Workshop on Statistical Machine Translation*, 376–380.

Ferreira, Flávio, Nuno Almeida, José Casimiro Pereira, Ana Filipa Rosa, André Oliveira & António Teixeira. 2013a. Multimodal and adaptable medication assistant for the elderly. Em *8th Iberian Conference on Information Systems and Technologies*, 309–314. Lisboa.

Ferreira, Flávio, Nuno Almeida, Ana Filipa Rosa, André Oliveira, José Casimiro Pereira, Samuel Silva & António Teixeira. 2013b. Elderly centered design for interaction - the case of the S4S medication assistant. Em *Proceedings of DSAI, Procedia Computer Science*, 398–408. Vigo.

Fonseca, Ana Cristina de Sena Raposo Paiva. 1993. *Comunicação em Linguagem Natural para um Tutor Inteligente*: Universidade Técnica de Lisboa, Instituto Superior Técnico Lisboa. Tese de Mestrado.

Hall, Mark, Ian Witten & Eibe Frank. 2011. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann 3rd edn.

Hastie, Helen & Anja Belz. 2014. A comparative evaluation methodology for NLG in interactive systems. Em *9th International Conference on Language Resources and Evaluation*, 4004–4011.

Hunter, J., E. Reiter, S. G. S. (Yaji) & J. Yu. 2005. Sumtime. (Online, verificado em 23/07/2015). http://www.abdn.ac.uk/ncs/departments/computing-science/sumtime-317.php.

Hunter, James, Yvonne Freer, Albert Gatt, Ehud Reiter, Somayajulu Sripada, Cindy Sykes & Dave Westwater. 2011. BT-Nurse: computer generation of natural language shift summaries from complex heterogeneous medical data. *Journal of the American Medical Informatics Association : JAMIA* 18. 621–624.

IRSTLM. 2011. Irst language modeling toolkit. (Online, verificado em 23/07/2015). http://sourceforge.net/projects/irstlm/.

Jurafsky, Daniel & James H. Martin. 2009. *Speech and language processing*. Prentic Hall 2nd edn.

Koehn, Philipp. 2014. *Moses: Statistical machine translation system - user manual and code guide*.

Koehn, Philipp, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin & Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. Em *45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions.*, 177–180. Praga, Rep. Checa: ACL.

Kohavi, Ron. 1995. A study of cross-validation and bootstrap for accuracy estimation and model selection. Em *Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 2* IJCAI'95, 1137–1143. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

Konstantopoulos, Stasinos, Ion Androutsopoulos, Haris Baltzakis, Vangelis Karkaletsis, Colin Matheson, Athanasios Tegos & Panos Trahanias. 2008. INDIGO: Interaction with personality and dialogue enabled robots [system demonstration]. Em *18th European Conference on Artificial Intelligence*, Patras, Greece.

Langner, Brian. 2010. *Data-driven natural language generation: Making machines talk like humans using natural corpora*: School of Computer Science - Carnegie Mellon University. Tese de Doutoramento.

Langner, Brian & Alan W. Black. 2009. MOUNTAIN: A translation-based approach to natural language generation for dialog systems. Em *First International Workshop on Spoken Dialogue Systems Techology*, .

Law, Anna S., Yvonne Freer, Jim Hunter, Robert H. Logie, Neil McIntosh & John Quinn. 2005. A comparison of graphical and textual presentations of time series data to support medical decision making in the neonatal intensive care unit. *J. Clin. Monit. Comput.* 19. 183–194.

Lemon, Oliver. 2010. Learning what to say and how to say it: joint optimization of spoken dialogue management and natural language generation. *Computer Speech & Language* 25. 210–221.

McCauley, Lee, Sidney D'Mello, Loel Kim & Melaine Polkosky. 2008. MIKI: A case study of an intelligent kiosk avatar and its usability. Em N. Magnenat-Thalmann, L. C. Jain & N. Ichalkaranje (eds.), *New Advances in Virtual Humans*, 153–176. Springer.

Mendes, Mateus Daniel. 2004. *Relações lexicais na geração de língua natural*: Universidade de Coimbra - Portugal. Tese de Mestrado.

MOSES. 2014a. Moses - baseline system. Online. http://www.statmt.org/moses/?n=Moses.Baseline.

MOSES. 2014b. Phrase-based tutorial. Online. http://www.statmt.org/moses/?n=Moses.Tutorial.

MOSES. 2014c. Syntax tutorial. Online. http://www.statmt.org/moses/?n=Moses.SyntaxTutorial.

Novais, Elder M., Rafael L. Oliveira, Daniel B. Pereira & Thiago D. Tadeu. 2009. A testbed for Portuguese Natural Language Generation. Em *Seventh Brazilian Symposium in Information and Human Language Technology*, 154–157. São Carlos, São Paulo, Brasil.

Och, Franz Josef. 2011. Giza++ statistical translation models toolkit. (Online, verificado em 23/07/2015). https://github.com/moses-smt/giza-pp.

Oliveira, Hugo Gonçalo. 2012. PoeTryMe: a versatile platform for poetry generation. Em *Proceedings of the ECAI 2012 Workshop on Computational Creativity, Concept Invention, and General Intelligence* C3GI 2012, Montpellier, France.

Papineni, Kishore, Salim Roukos, Todd Ward & Wei jing Zhu. 2002. Bleu: a Method for Automatic Evaluation of Machine Translation. Em *Meeting of the Association for Computational Linguistics*, 311–318.

Pereira, José Casimiro, António Teixeira & Joaquim Sousa Pinto. 2015. Towards a Hybrid NLG System for Data2Text in Portuguese. Em *Proceedings da 10ª Conferência Ibérica de Sistemas e Tecnologias de Informação* CISTI 2015, 679–684. Águeda, Portugal.

Portet, François, Ehud Reiter, Albert Gatt, Jim Hunter, Somayajulu Sripada, Yvonne Freer & Cindy Sykes. 2009. Automatic generation of textual summaries from neonatal intensive care data. *Artificial Intelligence* 173. 789–816.

Reiter, Ehud. 2007. An architecture for data-to-text systems. Em *Proceedings of the Eleventh European Workshop on Natural Language Generation.*, 97–104. Association for Computational Linguistics.

Reiter, Ehud & Robert Dale. 1997. Building applied natural language generation systems. *Journal of Natural Language Engineering – Cambridge University Press* 3(1). 57–87.

Reiter, Ehud & Robert Dale. 2000. *Building Natural Language Generation Systems*. Cambridge University Press.

Ribeiro, António. 1995. *Natural Language Generation with Rhetorical Relations and Focus Theory*: Edinburgh University – UK. Tese de Mestrado.

Salzberg, Steven L. & Usama Fayyad. 1997. On comparing classifiers: Pitfalls to avoid and a recommended approach. *Data Mining and Knowledge Discovery* 317–328.

Silva Junior, Douglas Fernandes Pereira, Ivandré Paraboni & Eder Miranda Novais. 2013. Um Sistema de Realização Superficial para Geração de Textos em Português. *RITA - Revista de Informática Teórica e Aplicada - Instituto de Informática da Universidade Federal do Rio Grande do Sul – Brasil* 20(3). 31–48.

Soares, Alexsandro Santos. 2001. *Gramática de Unificação Funcional: Levantamento de Requisitos para a Geração Sentencial de Português*: Instituto de Ciências Matemáticas e da Computação - USP/São Carlos - Brasil. Tese de Mestrado.

Teixeira, António, Flávio Ferreira, Nuno Almeida, Ana Filipa Rosa, José Casimiro, Samuel Silva, Alexandra Queirós & André Oliveira. 2013a. Multimodality and adaptation for an enhanced mobile medication assistant for the elderly. Em *Third Mobile Accessibility Workshop (MOBACC), CHI 2013 Extended Abstracts*, Paris.

Teixeira, António, Alexandra Queirós & Nelson Pacheco Rocha (eds.). 2013b. *Laboratório vivo de usabilidade (Living Usability Lab)*. ARC Publishing.

Turner, Ross, Somayajulu Sripada, Ehud Reiter & Ian P. Davy. 2006. Generating spatio-temporal descriptions in pollen forecasts. Em *Proceedings of the Eleventh Conference of the European Chapter of the Association for Computational Linguistics: Posters & Demonstrations.*, 163–166. Trento, Itália: Association for Computational Linguistics.