Proceedings of Symposium in Information and Human Language Technology. Uberlândia, MG, Brazil, October 2–5, 2017.

Normalizador de Texto para Língua Portuguesa baseado em Modelo de Linguagem

Patrick Thiago Bard[1], **Renan Lopes Luis**[1], **Silvia Maria Wanderley Moraes**[1]

[1]Faculdade de Informática –Pontifícia Universidade Católica do Rio Grande do Sul
Caixa Postal 1429 – 90.619-900 – Porto Alegre – RS – Brasil

{patrickthiagobard,renanlopesluis}@gmail.com, silvia.moraes@pucrs.br

Abstract. *Automatic processing of user-generated content on the Internet is a major challenge. Informal writing is one reason for this difficulty. This informality motivated the research on methods for text normalization. Text normalization is a step that precedes the usual processing, converting the text from user into a 'standard' (more formal) writing format. In this work, we prototype a normalizer for the Portuguese Language that is based on language model. In this approach, we use the machine translation technique to normalize the texts. We tested our normalizer in a corpus on Politics and compared the results obtained with those of another normalizer.*

Resumo. *O processamento automático de textos gerados pelo usuário na internet têm sido um grande desafio. A escrita informal é uma das razões dessa dificuldade. Essa informalidade têm motivado a pesquisa por métodos para normalização de textos. A normalização de texto é uma etapa que precede o processamento usual, convertendo o texto gerado pelo usuário em um formato 'padrão' (mais formal). Neste trabalho, prototipamos um normalizador para a Língua Portuguesa que é baseado em modelo de linguagem. Nessa abordagem, usamos a técnica de tradução automática para normalizar os textos. Testamos nosso normalizador em um corpus sobre política e comparamos os resultados obtidos com os de outro normalizador.*

1. Introdução

Os avanços tecnológicos das últimas décadas propiciaram a criação de novos ambientes de comunicação virtual, nos quais o emprego de uma linguagem mais informal é uma prática muito comum. Os *chats* e as mídias sociais, por exemplo, seguem essa tendência. No âmbito social, a informalidade na escrita é tolerada, tornando-se aceitável o uso de expressões reduzidas (ex. 'vc' ao invés de 'você'); erros de ortografia, de pontuação e de concordância (ex: '... saiba escalr muda isso ai e poe gente que sabe jogar'); bem como a repetição de letras com fins de ênfase (ex: 'Goooooool'), além do uso de gírias e de expressões em outras línguas (ex: 'esse note é show'). Essa liberdade de escrita dificulta o processamento automático desses textos, sendo necessário um tratamento preliminar para os mesmos afim de viabilizar a extração correta das suas informações. A área que tem se preocupado com esse tratamento é conhecida como Normalização de Textos. Ela visa transformar a escrita 'informal' de um texto em uma forma 'padrão' (mais formal e mais adequada para uma determinada aplicação) [Duran et al. 2014].

A Normalização de Textos é útil em diversas aplicações. Ela é necessária, por exemplo, em sistemas de busca; de reconhecimento de fala; de diálogo; de análise de

conteúdo gerado por usuários na web; tradução automática; etc. É importante ressaltar que embora existam diversas ferramentas para o processamento e análise dos textos com altos níveis de acertividade, tais ferramentas não conseguem trabalhar adequadamente com textos da web gerados pelos usuários. Mesmo quando conseguem, há uma redução significativa da acertividade. Isso acontece, principalmente, porque tais ferramentas foram definidas ou treinadas a partir de textos jornalísticos, nos quais a escrita é mais formal, ou seja, procura seguir de forma mais fiel a gramática da língua.

Neste trabalho propomos e analisamos um normalizador de textos baseado em modelo de linguagem. O objetivo é usar técnicas de tradução de texto, mas com a finalidade de normalizar textos em uma mesma língua. Semelhante à tradução automática de texto, usamos um *corpus* paralelo para treinar o normalizador. O *corpus* usado contém duas versões dos mesmos textos: uma contendo os textos originais (sem qualquer correção gramatical) e outra contendo os textos normalizados (traduzidos para uma forma mais padrão de escrita). O *corpus* usado foi chamado de Impeachment-BR e possui 500 *tweets* em português. Esses *tweets* foram coletados durante o processo de admissão do *Impeachment* da ex-Presidente Dilma Rousseff na câmara dos deputados.

O normalizador foi desenvolvido para a língua portuguesa, principalmente, porque os estudos nessa área para essa língua ainda são recentes. Encontramos apenas um normalizador de textos para o português, o UGCNormal [Duran et al. 2015]. O UGCNormal segue uma abordagem baseada em léxico, na qual são aplicadas várias regras de transformação (reescrita) ao texto. Embora o normalizador UGCNormal, nesse estudo, tenha obtido resultados melhores, consideramos a abordagem proposta promissora. Acreditamos, baseados em nossa análise, que o tamanho reduzido do *corpus* testado foi determinante para tal desempenho.

2. Normalização de Texto

A Normalização de Texto é o processo no qual o formato de um texto é convertido em um formato considerado padrão [Jurafsky and Martin 2009, Duran et al. 2014]. Por padrão entende-se como o formato mais adequado para uma determinada aplicação. De acordo com [Duran et al. 2014], a normalização de texto pode variar conforme: o gênero do texto de entrada; o formato desejado de saída; o propósito da normalização, e o método utilizado para executar essa tarefa. É importante levar em consideração tais características para definir claramente o que a 'normalização de texto' significa em cada contexto.

Apesar dos avanços na área de linguística computacional sejam notáveis, podemos observar algumas deficiências quando o processamento envolve textos curtos escritos de forma mais livre, em que os padrões usuais de escrita não são respeitados. As técnicas típicas para processamento de texto estão preparadas para lidar com poucos gêneros de texto, em sua maioria, textos jornalísticos, que usam uma linguagem mais formal. Logo, como esperado, tais técnicas não provêem um bom resultado quando aplicadas a gêneros de texto mais informais, com estruturas de construção mais livres[Sproat et al. 2001]. Técnicas que se baseim em algoritmos de aprendizagem podem ser treinadas para trabalhar com esses novos gêneros de texto. Entretanto, o problema é que dados anotados para esse tipo de abordagem não estão prontamente disponíveis e são difíceis de serem criados. Uma das dificuldades é a rápida evolução da linguagem usada nos textos gerados pelos usuários na web [**?**]. O dinamismo da linguagem permite uma mudança contínua na

forma como as pessoas se expressam. A falta de *corpus* é um problema com o qual convivemos. Por isso, tivemos que criar um *corpus* paralelo que atendesse as necessidades de nosso estudo.

As abordagens para normalização de texto usualmente dividem-se em dois grupos [Schlippe et al. 2010]: baseadas em léxicos e baseadas em modelos de linguagem. A abordagem baseada em léxico é mais tradicional e trata o problema de normalização como uma sequência de subproblemas que devem ser resolvidos[Duran et al. 2015]. É comum o uso de um conjunto de regras de substituição que vão transformando palavras 'desconhecidas' (*Out-of-Vocabulary* - OOV) em suas formas padrões correspondentes. Esta abordagem é utilizada pela ferramenta UGCNormal [Duran et al. 2014]. Nesta ferramenta, inicialmente o texto é quebrado em sentenças e, posteriormente, em *tokens*. Na etapa seguinte de verificação ortográfica, os tokens são corrigidos. A ferramenta trata ainda acrônimos, gírias e nomes próprios.

Já nas abordagens baseadas em modelo de linguagem, a normalização é tratada como um problema de tradução e exige um *corpus* paralelo. Nessa abordagem, o texto informal é traduzido para uma forma padrão. Exige que as sentenças não normalizadas estejam alinhadas com aquelas que são as suas versões normalizadas. As etapas mais usuais nesse tipo de abordagem consistem em pré-processamento, alinhamento e treinamento. A etapa de pré-processamento dos textos é responsável por limpar e uniformizar a tipografia do texto (caixa alta ou baixa), bem como por segmentá-lo em sentenças e, posteriormente, em termos[1]. Nessa etapa pode ser usado um analisador morfológico para detectar números e datas, bem como para reconhecer termos compostos e nomes próprios. Pode ser incluído também algum processamento de natureza semântica para desambiguação de sentido. Na etapa de alinhamento, os textos não normalizados e normalizados são perfilados, tornando a tradução viável. Esse alinhamento pode ser 'um-para-um' (correspondência direta entre palavras do texto não normalizado com as do normalizado), 'nulo-para-um' (a palavra não normalizada não tem influência no texto e é descartada) ou 'muitos-para-um' (uma sequência de termos - uma expressão- na versão não normalizada corresponde ao significado de uma única palavra da versão normalizada). E, por fim, na etapa de treinamento, os dados alinhados são usados para ensinar o tradutor. A coocorrência de palavras e frases nesses dados costuma ser usada para inferir correspondências de tradução entre duas línguas de interesse ou, no nosso caso, entre as formas de escrita dos textos normalizado e não normalizado.

Usamos em nosso estudo ferramentas estatísticas de tradução automática. Logo , a abordagem investigada é independente de linguagem.

3. Trabalhos Relacionados

Como já mencionado, UGCNormal foi um dos poucos normalizadores de texto que encontramos para a Língua Portuguesa. Não é de nosso conhecimento a existência de normalizadores baseados em modelo de linguagem para este idioma. Sendo assim, nessa seção descrevemos normalizadores que seguem a abordagem baseada em modelo de linguagem, mas construídos para outras línguas. Schilippe et al em [Schlippe et al. 2010] tratam a normalização de texto como um problema de tradução. O estudo teve como

[1]Termos podem ser símbolos, palavras ou n-gramas (sequência contínua de tokens com comprimento igual a n)

alvo a língua francesa. Nele foi usada a ferramenta Moses[2], sendo que o alinhamento do texto foi realizada pela ferramenta GIZA++[3] e o modelo de linguagem gerada pela ferramenta SRILM[4]. Os autores notaram que enquanto as normalizações manuais feitas pelos falantes nativos levaram cerca de 11 horas durante 3 dias, o normalizador foi melhorando o seu tempo de processamento. As primeiras 100 sentenças foram normalizadas em 114 minutos; o segundo grupo de 100 sentenças em 92 minutos e o terceiro grupo em apenas 10 minutos. Isso gerou uma média de 39,48 segundos por sentença. Ludena et al [Lopez Ludeña et al. 2012] também propuseram uma arquitetura baseada em tradução automática para normalizar textos, mas para a língua inglesa. Essa arquitetura era composta por: um módulo tokenizador responsável por segmentar o texto de entrada e transformá-lo em um grafo de tokens; um módulo tradutor que convertia os tokens para uma linguagem alvo e verificava se havia palavras fora do padrão (OOV) e, por fim, um módulo de pós-processamento para remoção de tokens desnecessários. Os autores igualmente usaram o Moses, sendo que Giza++ como alinhador e SRILM para geração do modelo de linguagem. Os resultados do normalizador foram considerados satisfatórios.

Na seção seguinte descrevemos os *corpora* utilizados em nosso estudo.

4. *Corpora* usados

Foram utilizados 2 *corpora* nessa investigação: Impeachment-BR e o Computer-BR. O primeiro é um *corpus* paralelo, que é o alvo de nosso estudo em normalização. E o segundo foi usado para melhorar o desempenho do normalizador, provendo mais termos.

4.1. *Corpus* Impeachment-BR

O desenvolvimento de um normalizador baseado em um modelo de linguagem exige um *corpus* paralelo. Nesse *corpus* devem existir duas versões dos mesmos textos: uma normalizada e outra não normalizada. Como não conhecíamos um *corpus* desse tipo para a língua portuguesa, foi parte do nosso estudo a construção de um. Para isso, trabalhamos sobre um subconjunto dos 157.420 *tweets*, em português, que foram coletados no dia 17 de abril de 2016. Os *tweets* eram sobre a votação da admissão do *impeachment* da ex-presidente Dilma Roussef. Como a normalização desses *tweets* seria manual e o *corpus* era muito grande, optamos, inicialmente, por anotar apenas as mensagens postadas durante o horário da votação, ou seja, entre 13h30 e 16h. No entanto, esse recorte resultou em 20 mil *tweets*. Decidimos, então, por reduzir ainda mais esse número e normalizamos, preliminarmente, 500 *tweets*. Esses *tweets* formam o *corpus* Impeachment-BR. A Tabela 1 apresenta um exemplo de *tweets* paralelos extraídos do *corpus* Impeachment-BR. Cabe mencionar que os *tweets* foram normalizados de forma colaborativa. Construímos uma ferramenta web especialmente para esse fim. Para que as contribuições fossem de algum modo padronizadas e houvesse poucas divergências quanto à forma de normalização, disponibilizamos um guia para os anotadores. Infelizmente, não tivemos a colaboração esperada no processo de normalização. Por essa razão, ele contou com apenas três anotadores que eram falantes nativos da língua portuguesa.

[2]https://github.com/moses-smt/mosesdecoder/tree/RELEASE-2.1.1
[3]https://github.com/moses-smt/giza-pp
[4]http://www.speech.sri.com/projects/srilm/

Corpus Informal	*Corpus* Formal
Vcs n tão entendendo	Vocês não estão entendendo
Eu queria ta em Brasilia agr	Eu queria estar em Brasília agora
#RespeiteAsUrnas Não vai ter golpe!Vai ter luta!	Respeite as urnas! Não vai ter golpe!Vai ter luta!

Tabela 1. Trecho do *corpus* Impeachment-BR

4.2. *Corpus* Computer-BR

O *corpus* Computer-BR é do domínio de Tecnologia e foi utilizado para otimizar (tuning) o normalizador. A função desse *corpus* de otimização é expandir os termos conhecidos, não deixando o normalizador restrito apenas aos termos que aparecem no *corpus* Impeachment-BR. O *corpus* Computer-BR possui 2.317 *tweets* em português, extraídos do Twitter durante o ano de 2015 [Moraes et al. 2016]. Não é um *corpus* paralelo e ele foi construído para estudos na área de Análise de Sentimentos. O uso desse *corpus* contribuiu para reduzir as OOV (palavras fora da língua padrão), melhorando o desempenho do normalizador. Do *corpus* Computer-BR foram usados apenas 200 *tweets*, os quais foram normalizados manualmente também.

5. Arquitetura do Normalizador

O normalizador foi implementado usando o framework Moses. O Moses é um sistema integrado de ferramentas de natureza estatística para o processo de tradução de máquina. A Figura 1 apresenta a arquitetura do normalizador proposto. Inicialmente, na fase de pré-processamento, os textos são tokenizados e têm sua tipografia normalizada (Truecasing) pelo Moses. Essa etapa é necessária tanto para o treinamento do normalizador quanto para o seu uso. Na fase de treinamento, foram usadas as ferramentas KenLM[5] para gerar um modelo de linguagem baseado em *n*-gramas, e MGIZA[6] para alinhar, em nível de *n*-gramas, os textos correspondentes do *corpus* paralelo Impeachment-BR. A etapa de Tuning é executada pelo Moses e faz uso do *corpus* Computer-BR.

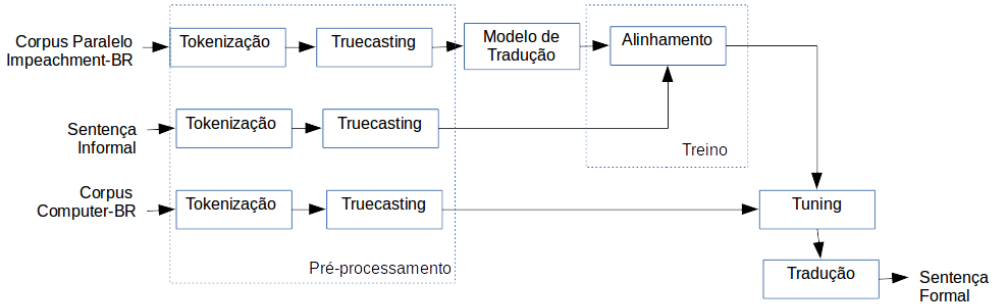

Figura 1. Arquitetura utilizada para o normalizador

[5]http://kheafield.com/code/kenlm/
[6]https://github.com/moses-smt/mgiza

6. Análise dos Resultados

Nós realizamos dois tipos de análise: uma quantitativa, que procura medir a acertividade do normalizador e outra qualitativa, que visa uma avaliação intrínseca dos resultados. Em nossa análise, usamos a métrica *bilingual evaluation understudy* (BLEU)[Papineni et al. 2002], que é bem usual na área de tradução automática.

6.1. Análise Quantitativa

Nessa análise, testamos 3 configurações de conjuntos de treino e teste (ver Tabela 2). Os conjuntos de cada configuração foram gerados aleatoriamente. A diferença entre as configurações é a quantidade de *tweets* no conjunto de treino. A cada nova configuração são acrescidos 50 *tweets* ao conjunto de treino. Nosso objetivo, nesse caso, era verificar se o tamanho do *corpus* de treino influenciava nos resultados.

Caso de Teste	*#Tweets* para treino	*#Tweets* para teste
1	350	150
2	400	100
3	450	50

Tabela 2. Casos de teste e a proporção de *tweets* utilizada

Usamos validação cruzada *k*-fold, onde *k=10*, portanto foram executadas 10 configurações diferentes de conjuntos de treino e teste em nosso normalizador. A Figura 2 apresenta o resultado médio da medida BLEU referente a essas execuções. Para fins de comparação o arquivo de teste de cada configuração analisada foi testado também no normalizador UGCNormal.

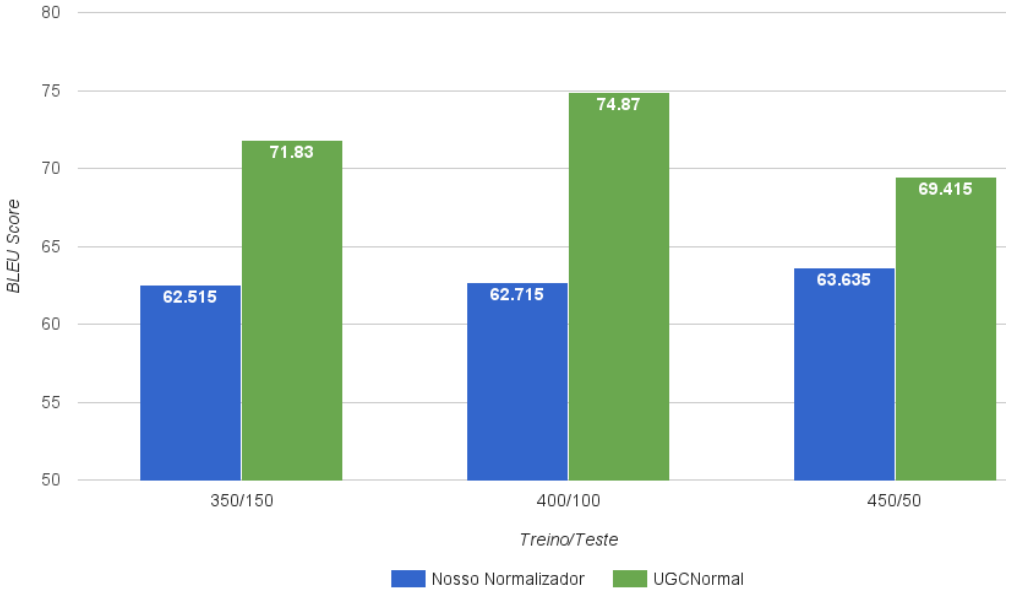

Figura 2. Comparativo entre os normalizadores estudados

O UGCNormal obteve resultados melhores que o nosso normalizador. Uma das razões é certamente o tamanho do *corpus* Impeachment-BR. Como a abordagem usada pelo Moses é estatística, uma frequência irrelevante de termos torna inexpressiva a possível correlação na qual esse termos estejam envolvidos. Como o *corpus* Impeachment-BR é relativamente pequeno, vários termos apresentaram baixa frequência, o que justifica o desempenho apresentado. Já o UCGNormal por ser baseado em regras, consegue tratar de forma satisfatória esses mesmos casos. Por outro lado, é possível notar que os resultados para o nosso normalizador vão crescendo lentamente a cada nova configuração testada. Isso indica que de fato existe uma tendência na obtenção de resultados cada vez melhores a medida que o tamanho do *corpus* utilizado aumenta. Cabe mencionar que essa expectativa de melhora não é esperada para o UCGNormal.

6.2. Análise Qualitativa

Nessa análise, avaliamos alguns casos de normalização de forma intrínseca a fim de determinar os pontos fortes e fracos da abordagem usada. Analisando algumas sentenças, observamos diferenças entre as abordagens baseada em modelo de linguagem (nosso normalizador) e baseada em léxico (UGCNormal). Por exemplo, para a sentença 'Vcs n tão entendendo', os normalizadores geraram saídas diferentes . Ambos normalizadores, transformaram 'vcs' em 'vocês', no entanto o verbo 'tão' foi mantido pelo UGCNormal, mas alterado para 'estão' no caso do nosso normalizador. Modificação semelhante ocorreu na sentença 'Não vai rolar, essa roubalheira tem que acabar ... '. Nosso normalizador, também trocou o verbo, substituindo 'rolar' por 'acontecer'. Em aplicações que exigem tratamento semântico, as transformações providas pelo nosso normalizador podem ser mais convenientes, pois a abordagem estatística garante que o termo gerado seja o mais frequente. Sendo o mais frequente, ele terá uma grande chance de corresponder à forma mais comum de sua escrita cujo significado também é mais usual. Isso contribui para redução de ambiguidade. Por exemplo, as normalizações geradas pelo UCGNormal 'tão' e 'rolar' podem ser confundidas com um advérbio e com a expressão 'fazer girar', respectivamente. Já no caso dos termos 'estão' e 'acontecer' a ambiguidade no significado é menor.

Observamos também que o UGCNormal não conseguiu tratar adequadamente alguns nomes próprios. Por exemplo, a sentença 'A deputada mariadorosario condena as tentativas de impeachment contra dilmabr' foi convertida em 'A deputada mariadorosario condena as tentativas de impeachment contra filmar'. O normalizador não conseguiu decompor o nome próprio 'Maria do Rosário' e, ainda, substituiu 'Dilma' incorretamente por 'filmar'. Nosso normalizador não produziu a transformação mais adequada, no entanto foi mais coerente ao gerar a saída. Ele produziu como saida: 'a deputada do Rosário condena as tentativas de impeachment contra Dilma'.

Já, no caso da sentença 'Eu queria ta em Brasília agr', o normalizador UGCNormal foi melhor. Nosso normalizador não conseguiu transformar 'agr' em 'agora'. Havia poucas ocorrências do termo 'agr' nos *corpora* usados. Logo, o normalizador acabou preservando o termo integralmente, dado que o alinhamento entre os termos 'agr' e 'agora' não existia.

7. Conclusão

Apesar do nosso normalizador de texto baseado em modelo de linguagem não ter gerado transformações melhores que as do UGCNormal, seus resultados são promissores. Acreditamos que o desempenho apresentado foi uma consequência do tamanho reduzido do *corpus* Impeachment-BR. Acreditamos também que a abordagem baseada em modelo de linguagem é mais adequada para acompanhar o dinamismo da língua natural, pois exige menos esforço quanto à atualização do normalizador. Para novas formas de escrita, basta treinar o normalizador novamente. Por outro lado, a ausência de *corpora* paralelos para a tarefa de normalização, principalmente para a língua portuguesa, ainda é um problema com o qual precisamos conviver. Por essa razão, consideramos o *corpus* Impeachment-BR uma de nossas contribuições. Como trabalhos futuros, pretendemos estender o *corpus*, bem como testar a abordagem para outros domínios.

8. Agradecimento

Nosso agradecimento a PUCRS (EDITAL N. 01/2016 - Programa de Apoio à Atuação de Professores Horistas em Atividades de Pesquisa) pelo apoio financeiro.

Referências

Duran, M. S., Avanço, L. V., Aluísio, S. M., Pardo, T. A. S., and Nunes, M. d. G. V. (2014). In proceedings of the 9th web as corpus workshop (wac-9). In *Some Issues on the Normalization of a Corpus Products Reviews in Portuguese*, pages 22–28, Washington, DC, USA. Association for Computational Linguistics.

Duran, M. S., Avanço, L. V., Nunes, M. d. G. V., et al. (2015). A normalizer for ugc in brazilian portuguese. In *Workshop on Noisy User-generated Text*. Association for Computational Linguistics-ACL.

Jurafsky, D. and Martin, J. H. (2009). *Speech and Language Processing*. Prentice-Hall, Inc., 2th edition.

Lopez Ludeña, V., San Segundo Hernández, R., Montero Martínez, J. M., Barra Chicote, R., and Lorenzo Trueba, J. (2012). Architecture for text normalization using statistical machine translation techniques. In *IberSPEECH 2012*, pages 112–122, Madrid, Spain. Springer.

Moraes, S. M. W., Santos, A. L. L., Redecker, M., Machado, R. M., and Meneguzzi, F. R. (2016). Comparing approaches to subjectivity classification: A study on portuguese tweets. In *Computational Processing of the Portuguese Language: 12th International Conference, PROPOR 2016*, pages 86–94, Tomar, Portugal. Springer International Publishing.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 311–318, Stroudsburg, PA, USA. Association for Computational Linguistics.

Schlippe, T., Zhu, C., Gebhardt, J., and Schultz, T. (2010). Text normalization based on statistical machine translation and internet user support. In *INTERSPEECH 2010, 11th Annual Conference of the International Speech Communication Association, Makuhari, Chiba, Japan, September 26-30, 2010*, pages 1816–1819.

Sproat, R., Black, A. W., Chen, S., Kumar, S., Ostendorf, M., and Richards, C. (2001). Normalization of non-standard words. *Comput. Speech Lang.*, 15(3):287–333.