SEMPLICe: Um Modelo Sequencial de Proficiência em Comunidades Online para Aprendizado de Idioma *

Rafael Sales Medina, Ana Paula Couto da Silva, Fabricio Murai

[1]Departamento de Ciência da Computação
Universidade Federal de Minas Gerais (UFMG) – Belo Horizonte, MG – Brazil

{rafael.medina, ana.coutosilva, murai}@dcc.ufmg.br

Abstract. *Reddit is an online social network where users interested in a common subject may interact with each other through subreddits. Subreddits for language learning have been attracting users of various proficiency levels each year, interested in boosting their learning. In particular, on subreddit German, users are advised to inform their proficiency level when writing a post. Yet, only 20% of the posts have such tags. In this paper we address the problem of classifying users' proficiency from their publications. We conduct experiments which show that classifiers that treat publications as independent observations perform poorly. We then propose a new model dubbed SEMPLICe, which uses both textual features and the publication history of an user to classify her proficiency level over time. By assuming that proficiency is monotonically non-decreasing as long as the user remains active, SEMPLICe yields a weighted F_1 score up to 29.6% higher than previous methods. SEMPLICe uses dynamic programming to achieve linear complexity on each user's history size.*

Resumo. *O Reddit é uma rede social online em que usuários interessados em um mesmo tópico interagem uns com os outros em subreddits. Subreddits para aprendizado de idioma vem atraindo usuários de diferentes níveis de proficiência a cada ano, buscando melhorar o aprendizado. Em particular, no subreddit German, os usuários são aconselhados a informar seu nível de proficiência ao escrever um post. Contudo, apenas 20% dos posts possuem tais tags. Abordamos aqui o problema de classificar a proficiência dos usuários a partir de suas publicações. Conduzimos uma série de experimentos que demonstram que classificadores que tratam as publicações como observações independentes tem baixo desempenho. À vista disso, propomos um novo modelo, SEMPLICe, que considera as características textuais e também o histórico de um usuário no subreddit para classificar sua proficiência ao longo do tempo. Baseado na suposição de que a proficiência é não decresce desde que um usuário permaneça ativo, SEMPLICe alcança um F_1 ponderado até 29,6% maior que os métodos anteriores. SEMPLICe utiliza programação dinâmica para obter complexidade linear no tamanho do histórico de cada usuário.*

1. Introdução

O Reddit é uma rede social online onde usuários interessados em um tema comum se registram em uma comunidade (*subreddit*), onde podem compartilhar conteúdo relacionado

*Este trabalho foi parcialmente financiado por recursos do CNPq, CAPES e FAPEMIG.

aquele tema (e.g., links, textos, imagens) na forma de *posts*, que iniciam *threads* de discussão. Estes posts podem ser comentados e avaliados por outros usuários. Em particular, subreddits para aprendizado de idiomas específicos vem atraindo cada vez mais usuários, e permitem que pessoas dos mais diversos níveis de proficiência compartilhem dúvidas e dicas de como melhorar o aprendizado, por exemplo.

Em outras pesquisas, como [Yang et al. 2016] e [Crossley et al. 2012], foi constatado que é possível classificar a proficiência de usuários a partir de textos de voluntários. Sabendo disso, nesta pesquisa levantamos a seguinte questão: é possível classificar com acurácia a proficiência de um usuário de subreddits a partir das características textuais extraídas de suas publicações? Abordamos esta questão de duas maneiras diferentes.

A primeira abordagem consiste em usar as publicações como observações independentes no treinamento de diversos modelos de classificação supervisionados (KNN, *Random Forest*, *Logistic Regression* e *Gradient Boosting*). Cada observação é composta por mais de características textuais extraídas pelo LIWC (*Linguistic Inquiry and Word Count*). O LIWC é uma ferramenta para análise psicolinguística que extrai 81 características textuais de cada publicação, como quantidade de palavras comuns, quantidade de pontuações, pronomes, artigos e preposições, e orientação temporal.

Esta abordagem não leva em consideração a identidade dos usuários que escreveram as publicações, tampouco a ordem em que elas apareceram. Já a segunda abordagem consiste na proposta de um modelo probabilístico sequencial que utiliza os classificadores treinados na primeira abordagem, mas considera a ordem das publicações realizadas por cada usuário e retorna os rótulos de proficiência que maximizam a verossimilhança da sequência. O modelo é baseado na premissa de que, para usuários que não ficam inativos durante períodos prolongados de tempo, a proficiência é uma função monotonicamente não-decrescente. Esse modelo foi denominado SEMPLICe (SEquential Model for Proficiency cLassifICation).

Para fins de claridade nas explicações, descrevemos primeiro a versão do modelo que testa todos os possíveis momentos em que a proficiência aumenta através de um algoritmo de força bruta, cuja complexidade é $\Theta(N^2)$, sendo N o número de publicações de um usuário. Em seguida, desenvolvemos uma versão do modelo baseada em um algoritmo de programação dinâmica, reduzindo a complexidade para $\Theta(N)$.

Os resultados foram obtidos a partir dos dados coletados em um *dump online* do Reddit. Este conjunto de dados é composto por todas as publicações (posts e comentários) de todos os usuários que participaram do subreddit *German*. Utilizando a primeira abordagem (não-sequencial), o *Gradient Boosting* apresentou resultados melhores que os outros modelos, tendo sido classificado como o melhor, com um valor de acurácia medido pela F_1 ponderada em torno de 0,45. Os resultados desse modelo foram utilizados para alimentar o SEMPLICe, que aumentou a acurácia para aproximadamente 0,60.

As análises realizadas nesse trabalho permitiram a melhoria nos resultados de classificação da proficiência dos usuários ao longo do tempo, o que se deve à criação do modelo sequencial. Esse resultado é de grande valia, dado que auxilia no acompanhamento da evolução de usuários em um idioma. Dessa maneira, um usuário pode acompanhar a evolução de sua proficiência e até mesmo prever quando chegará no próximo nível baseado em suas interações. Além disso, professores do idioma podem utilizar para

acompanhar a evolução de alunos. Finalmente, os resultados apresentados neste artigo podem servir como base para a criação de aplicativos e mídias sociais específicos e mais eficazes para o aprendizado de línguas por meio da interação entre pessoas utilizando uma rede virtual.

Este artigo está organizado da seguinte forma: a Seção 2 apresenta os principais trabalhos em *Assisted Language Learning* e classificação automática de proficiência; a Seção 3 descreve o dataset e características textuais utilizadas; as Seções 4 e 5 detalham a abordagem não-sequencial e o SEMPLICe, respectivamente; os experimentos e resultados são detalhados na Seção 6; por fim, a conclusão e as limitações do projeto são apresentadas na Seção 7.

2. Trabalhos Relacionados

O aprendizado de línguas apoiado por tecnologias, campo de estudos conhecido por *Computer Assisted Language Learning* (CALL) [Levy 1997], tem sido estudado há mais de duas décadas [Zhao 1996, Warschauer and Healey 1998]. A CALL engloba todos os tipos de ferramentas computacionais que possam auxiliar no aprendizado de um segundo idioma, sendo as redes sociais online um exemplo que tem sido cada vez mais investigado por pesquisadores, como discutido em [Zourou 2012]. Essa pesquisa demonstra como as redes sociais online influenciam a participação dos usuários, e têm impacto positivo no ensino de idiomas.

Corroborando essa conclusão, [Arnold and Paulus 2010] demonstra como um fórum criado para o apoio ao aprendizado de um idioma tem impacto positivo no aprendizado, tanto sob o ponto de vista daquele que ensina, quanto daquele que aprende. Porém, na pesquisa de [Lin et al. 2016], é reforçada a necessidade de que as redes ofereçam apoio e orientação ao usuário para incentivar o engajamento e, como consequência, promover o desenvolvimento do aprendizado de idioma. Outras pesquisas que focam em comunidades online de aprendizado de línguas demonstram que, em comunidades do Reddit voltadas para esse tópico, os usuários têm interesse pelo conteúdo compartilhado, mais que nas interações interpessoais. Além disso, também demonstrou-se que (i) os usuários de diferentes níveis de proficiência interagem entre si e que (ii) os textos de cada um desses grupos tem características específicas. Esta segunda observação é uma evidência da possibilidade de classificá-los de acordo com a proficiência.

Características textuais podem ser utilizadas para classificação automática de proficiência, como demonstra a pesquisa de [Yang et al. 2016], que utiliza resultados de testes de inglês dos mais diversos tipos, como escrita e leitura, para medir a aptidão cognitiva em um segundo idioma de um grupo de voluntários. Os resultados desses testes são utilizados para alimentar modelos de aprendizado de máquina, como Regressão Logística e *Random Forest*, que com uma precisão de 70% conseguiram classificar corretamente a proficiência dos voluntários. A pesquisa de [Crossley et al. 2012], por sua vez, realiza um trabalho similar de análise de proficiência em inglês, porém utilizando apenas textos como dados de entrada. Esses textos foram analisados com o apoio da ferramenta Coh-Metrix, que quantifica, entre outros pontos, a coesão de textos em inglês. Esse trabalho demonstra como textos de níveis de proficiência diferentes apresentam características diferentes.

O presente trabalho apresenta uma nova proposta de avaliação automática da proficiência em um segundo idioma, já que realiza esta análise a partir de textos publica-

dos em redes sociais. Os textos de usuários são analisados por meio da ferramenta de análise textual LIWC que, diferente das ferramentas utilizadas em outros trabalhos, permite a análise dos mais diversos idiomas, incluindo português, alemão e espanhol. Um outro ponto que difere este trabalho é que ao contrário dos trabalhos mencionados anteriormente, que testaram a proficiência de voluntários, utilizamos apenas textos reais publicados em uma comunidade online. Além disso, propomos um modelo que utiliza o histórico dos textos de um mesmo usuário para calcular a proficiência, ao contrário de modelos anteriormente mencionados.

3. Dataset e extração de características

O Reddit possui seus dados, incluindo todas as publicações em subreddits, disponíveis publicamente na web[1]. Foram coletadas todas as 159.407 publicações de usuários do subreddit *German* entre janeiro de 2010 e dezembro de 2017. Essa comunidade foi escolhida por recomendar explicitamente a seus usuários que indiquem seu nível de proficiência, resultando em número 29.806 publicações com a proficiência auto-declarada. Dessa maneira, foram selecionadas apenas as publicações que incluem a indicação da proficiência, excluindo usuários nativos, visto que a proposta do presente trabalho é modelar exclusivamente a proficiência de usuários que estão aprendendo um novo idioma. Foram extraídos 9.569 publicações de iniciantes, 12.211 de intermediários e 8.026 de avançados.

Os textos dessas publicações foram então analisados pela ferramenta LIWC, que quantifica cerca de 80 características para cada um dos textos. Essas características, todas representadas por valores númericos, incluem atributos afetivos, atributos cognitivos e atributos de estilo linguístico, e foram utilizadas para treinar métodos de classificação supervisionada. Este é um problema de classificação multi-classe, pois um usuário pode ser classificado como iniciante, intermediário e avançado com relação a sua proficiência.

4. Modelagem Não-Sequencial para Classificação de Proficiência

Nesta abordagem, denominada **não-sequencial**, a predição da proficiência é feita a partir de cada publicação (post ou comentário) de maneira independente, não levando em consideração a identidade do usuário, nem a ordem em que elas foram realizadas.

Denotamos por x_i o vetor de *features* de uma publicação i e y_i a proficiência associada a ela. Treinamos diversos classificadores a partir do conjunto de dados para modelar a relação entre as features x_i (extraídas pelo LIWC) e a proficiência y_i. Utilizamos classificadores clássicos para aprendizado supervisionado multi-classe: (1) KNN (*K-Nearest-Neighbours*) [Altman 1992], (2) *Random Forest* [Breiman 2001], (3) *Logistic Regression* [Yu et al. 2011], e (4) *Gradient Boosting* [Friedman 2002].

Com a finalidade de determinar os parâmetros a serem usados por cada modelo, utilizamos o método Grid Search, que seleciona os valores ótimos para os parâmetros baseado em um conjunto de dados de validação amostrado a partir do conjunto de dados de treinamento. Então, para cada um dos modelos e suas respectivas parametrizações, foram realizados novos experimentos para compará-los entre si e definir qual dentre eles apresentou melhor desempenho.

[1] http://files.pushshift.io/reddit/

Nesta abordagem, a proficiência do usuário em um determinado instante é estimada apenas a partir das características textuais de sua última publicação. Portanto, não inclui nenhum mecanismo para forçar a consistência na evolução da proficiência de um dado usuário. Por exemplo, o classificador pode prever que a proficiência associada a um post i é menor que aquela associada a outro post escrito pelo mesmo usuário em um momento imediatamente anterior. Para contornar esse problema, foi proposto um novo modelo, que considera não apenas as características do texto no momento de publicação, como também o histórico de publicações, de maneira que é respeitada a suposição da evolução da proficiência ser constante ou crescente. Esse modelo é apresentado com maior detalhamento na próxima seção.

5. SEMPLICe: um modelo sequencial de proficiência

Nesta seção iremos propor um modelo probabilístico que resolve a principal deficiência da abordagem anterior, que é não considerar o histórico de publicações na classificação da proficiência de um usuário. Denominamos o modelo SEMPLICe (SEquential Model for Proficiency cLassifICation), que significa *simples*, em italiano.

Desejamos encontrar a sequência de níveis de proficiências $Y_1 \leq Y_2 \leq \ldots \leq Y_N$, que maximiza a probabilidade condicional $P(Y_1, \ldots, Y_N | X_1, \ldots, X_N)$. SEMPLICe é baseado em duas suposições. A primeira suposição é de que o nível de proficiência é monotonicamente crescente, ou seja, permanece igual ou aumenta de Y_t para Y_{t+1}. Esta suposição é razoável se considerarmos usuários que não permanecem inativos durante muito tempo. A segunda suposição é de que, condicionado à restrição de que Y_t é não-decrescente, a probabilidade $P(Y_1, \ldots, Y_n | X_1, \ldots, X_N)$ pode ser aproximada por um produto de fatores:

$$P(Y_1, \ldots, Y_n | X_1, \ldots, X_N) \approx \prod_{t=1}^{N} P(Y_t | X_t), \quad \text{onde } Y_1 \leq Y_2 \leq \ldots Y_N. \tag{1}$$

A base lógica para a segunda suposição é de que probabilidades de transição do tipo $P(Y_{t+1}|Y_t)$ variam de usuário para usuário conforme fatores externos, não podendo ser aprendidas de forma efetiva a partir do conjunto de dados que utilizamos. Embora seja possível usar o estimador frequentista, tal estimador representa uma média de uma população potencialmente muito diversa. Note que esta suposição não torna independentes as classificações de publicações de um mesmo usuário, visto que elas devem satisfazer à primeira suposição (i.e., $Y_1 \leq Y_2 \leq \ldots Y_N$).

O problema de se encontrar a sequência de rótulos $\mathbf{Y} = \{Y_t\}_{t=1}^{N}$ que maximiza o lado direito de (1) é, após a transformação logarítmica, equivalente ao problema de otimização

$$\max_{\mathbf{Y}: Y_1 \leq Y_2 \leq \ldots \leq Y_N} \sum_{i=1}^{N} \log P(Y_t | X_t). \tag{2}$$

As probabilidades $P(Y_t|X_t)$ são calculadas a partir dos modelos de classificação treinados na abordagem não-sequencial. Primeiro, é realizada a etapa de treinamento do modelo, que é realizada com uma parte do conjunto total de publicações. Então, na etapa de predição o modelo recebe como entrada as *features* X_t de cada publicação t, e

Algoritmo 1: Algoritmo sequencial: versão força bruta

entrada: matriz $\mathbf{M}_{3\times N}$, onde $M[i,t] = \log P(Y_t = i|X_t)$
saída : par TRANSICOES $= (t_i, t_a)$ indicando que a transição para intermediário acontece no passo t_i e para avançado, no passo t_a

```
/* Caso em que não há transições                                    */
```
1 TRANSICOES ← (Nunca, Nunca)
2 MAXLOGLIKELIHOOD ← $\sum_{t=1}^{N} M[1,t]$
3 **for** t_i *de* 1 *até* N **do**
```
    /* Caso em que se torna intermediate em t_i                     */
```
4 LOGLIKELIHOOD ← $\sum_{t=1}^{t_i-1} M[1,t] + \sum_{t=t_i}^{N} M[2,t]$
5 **if** LOGLIKELIHOOD > MAXLOGLIKELIHOOD **then**
6 MAXLOGLIKELIHOOD ← LOGLIKELIHOOD
7 TRANSICOES ← $(t_i$, Nunca)
8 **for** t_a *de* $t_i + 1$ *até* N **do**
```
    /* Caso em que transições ocorrem em t_i e t_a                  */
```
9 LOGLIKELIHOOD ← $\sum_{t=1}^{t_i-1} M[1,t] + \sum_{t=t_i}^{t_a-1} M[2,t] + \sum_{t=t_a}^{N} M[3,t]$
10 **if** LOGLIKELIHOOD > MAXLOGLIKELIHOOD **then**
11 MAXLOGLIKELIHOOD ← LOGLIKELIHOOD
12 TRANSICOES ← (t_i, t_a)
13 **return** TRANSICOES

retorna um vetor $[p_b, p_i, p_a]$, que apresenta a probabilidade de que aquela publicação seja classificada em cada nível de proficiência.

Para resolver o problema de otimização em (2) propomos um algoritmo de programação dinâmica na Seção 5.2, No entanto, para um melhor entendimento, descrevemos primeiro um algoritmo de força bruta cuja complexidade é $\Theta(N^2)$ na Seção 5.1.

5.1. Algoritmo I: Força Bruta

O Algoritmo 1 descreve em pseudocódigo a versão força bruta do algoritmo sequencial. Para estimar a sequência de rótulos para um usuário com N publicações, ela recebe como entrada uma matriz $\mathbf{M}_{3\times N}$, onde $M[i,t] = \log P(Y_t|X_t)$, onde $i \in 1, 2, 3$ representam os níveis de proficiência *beginner*, *intermediate* e *advanced*, respectivamente. A saída do algoritmo é um par TRANSICOES $= (t_i, t_a)$ indicando que o usuário passou a ter nível intermediário em t_i e avançado em t_a.

O algoritmo calcula o log-likelihood da sequência para todas as combinações possíveis de instantes de transição (de iniciante para intermediário e de intermediário para avançado). Isso é possível utilizando dois laços aninhados, onde o primeiro (t_i), define a publicação a partir da qual o usuário passa a ser intermediário (partindo de 1, i.e., nenhuma publicação iniciante). O segundo ($t_a > t_i$) define qual a primeira publicação em que o nível de proficiência do usuário é avançado.

Entre as linhas 4 e 7, consideramos o caso em que o usuário permanece no nível 2 (intermediário) até o fim da sequência. Entre as linhas 8 e 12, consideramos o caso em que fazer uma outra transição para o nível 3 (avançado) é mais provável. A complexidade computacional do algoritmo é $\Theta(N^2)$. Dentre as possíveis sequências, três apresentam apenas uma proficiência, $2\times(N-1)$ apresentam apenas uma transição e $(N-1)(N-2)/2$ apresentam duas.

Algoritmo 2: Algoritmo sequencial: versão programação dinâmica

entrada: matriz $\mathbf{M}_{3\times N}$, onde $M[i,t] = \log P(Y_t = i|X_t)$
saída : par TRANSICOES $= (t_i, t_a)$ indicando que a transição para intermediário acontece no passo t_i e para avançado, no passo t_a

```
/* Inicialização da última linha de R                                  */
R[3, N] ← M[3, N]
for t de N − 1 até 1 do
    R[3, t] = M[3, t] + R[3, t + 1]
TRANSICOES ← (Nunca, Nunca)
for p de 2 até 1 do
    for t de N até 1 do
        if t = N then
            /* Máximo, caso termine sequência em p                     */
            LOGLIKELIHOOD ← M[p, t]
        else
            /* Máximo, caso proficiência seja p em t                   */
            LOGLIKELIHOOD ← M[p, t] + R[p, t + 1]
        if R[p + 1, t] > LOGLIKELIHOOD then
            /* Trocar para p+1 em t é mais provável                    */
            R[p, t] ← R[p + 1, t]
            TRANSICOES[p] = t
        else
            /* Permanecer em p é mais provável                         */
            R[p, t] ← LOGLIKELIHOOD
return TRANSICOES
```

5.2. Algoritmo II: Programação dinâmica

O Algoritmo 2 descreve a versão do SEMPLICe baseada em programação dinâmica, que tem as mesmas entradas e saídas da versão força bruta, mas cuja complexidade é $\Theta(N)$ em vez de $\Theta(N^2)$. Esta versão se baseia na construção de uma memória de cálculo representada pela matriz $\mathbf{R}$, onde

$$\mathbf{R}[p,i] = \max_{p \le Y_i \le Y_{i+1} \le \ldots \le Y_N} \sum_{t=i}^{N} \log P(Y_t|X_t),$$

ou seja, o máximo da verossimilhança da subsequência $\{Y_i, \ldots, Y_N\}$, dado que $Y_i \geq p$.

Para calcular $\mathbf{R}[p,t]$, precisamos determinar se é mais provável transicionar para um nível de proficiência superior ou permanecer no nível p no instante t. No primeiro caso, o máximo da função de log-verossimilhança ao transicionar para $p+1$ imediatamente antes de fazer a publicação X_t é dado por $\mathbf{R}[p+1,t]$. No segundo caso, a máximo da função de log-verossimilhança é dado pela soma de $\mathbf{M}[p,t] = \log P(Y_t = p|X_t)$ e $\mathbf{R}[p,t+1]$. Portanto, para $p \in \{1,2\}$ e $t \in \{1,\ldots,N-1\}$, temos a relação de recorrência

$$\mathbf{R}[p,t] = \max\left(\mathbf{R}[p+1,t],\, \mathbf{M}[p,t] + \mathbf{R}[p,t+1]\right). \tag{3}$$

A partir da Eq. (3) observa-se que a matriz $\mathbf{R}$ pode ser construída de "baixo para cima" e, dentro de cada linha, da "esquerda para a direita". Falta, portanto, definir como a última linha e a última coluna serão inicializadas. Tendo em vista que o nível máximo de proficiência é 3 (avançado), inicializamos $\mathbf{R}[3,i] = \sum_{t=i}^{N} \mathbf{M}[3,t]$ entre as linhas 1 e 3 do pseudocódigo, ou seja, considerando que não haverá novas transições. Já para preencher a coluna $\mathbf{R}[p,N]$, fazemos

$$\mathbf{R}[p,N] = \max\left(\mathbf{R}[p+1,N],\, \mathbf{M}[p,N]\right). \tag{4}$$

Este algoritmo retorna exatamente o mesmo resultado que a versão força bruta. No entanto, a análise de complexidade desta variante demonstra que ela é $\Theta(N)$, representando uma redução significativa em relação à versão anterior, que é $\Theta(N^2)$, especialmente para sequências longas. A variante baseada em programação dinâmica é a que foi de fato utilizada para a modelagem dos dados reais.

6. Experimentos e resultados

A seguir descrevemos os resultados da classificação de proficiência obtidos pelas duas abordagens. Na Seção 6.1, avaliamos os modelos de classificação que não levam a sequência de publicações em consideração. Na Seção 6.2, utilizamos as previsões obtidas a partir dos modelos treinados na etapa anterior para alimentar o algoritmo sequencial. Os resultados desta abordagem são então comparados com aqueles da abordagem não-sequencial. Para realizar os testes, foram selecionadas para teste apenas as publicações de usuários que apresentaram sequências de evolução da proficiência sem inconsistências, ou seja, que se mantiveram constantes ou crescentes. Dessa maneira, de todo o conjunto de publicações, foram selecionadas 9099 publicações de teste de 1219 usuários diferentes que apresentaram sequências consistentes de proficiência.

Antes de treinar os classificadores, utilizamos o método chi-squared para seleção de features que, como apontado em [Spolaôr and Tsoumakas 2013], é o melhor método para classificação textual. Contudo, após avaliar combinações envolvendo pelo menos 5 features, o método não descartou nenhuma das 80 features. Por isso, foram utilizadas todas as features disponibilizadas.

Utilizamos a métrica F_1 ponderada para comparar os resultados de classificação entre métodos, que calcula o valor de F_1 para cada classe (iniciante, intermediário e avançado) e retorna a média ponderada pelo suporte (número de instâncias de cada classe). Essa métrica, comum para analisar a acurácia de modelos, é calculada a partir da média harmônica entre **revocação** (fração de instâncias relevantes que são recuperadas) e **precisão** (fração de instâncias recuperadas que são relevantes) dos resultados dos classificadores. Além disso, para entender quais são os tipos de erro cometidos por cada modelo, apresentamos também as respectivas matrizes de confusão.

6.1. Resultados de classificação usando abordagem tradicional (não-sequencial)

Cada um dos modelos de classificação utilizados nesta avaliação (especificamente, KNN, Random Forest, Logistic Regression e Gradient Boosting) possui hiper-parâmetros que precisam ser escolhidos antes que seja feito o ajuste aos dados. Para selecionar o valor de cada um dos hiper-parâmetros, utilizamos o método Grid Search [Bergstra and Bengio 2012].

Tabela 1. Média e desvio padrão do F_1 ponderado obtido pelos modelos não-sequenciais após tuning de hiper-parâmetros (10-fold cross-validation).

	Logistic Regression	Random Forest	KNN	Gradient Boosting
Média	0,3786	0,3995	0,4092	0,4568
Desvio	0,0150	0,0108	0,0089	0,0093

A Tabela 1 mostra a média e o desvio padrão da F_1 ponderada obtida por cada um dos modelos não-sequenciais. Os valores foram obtidos a partir de 10-fold cross-validation, método selecionado para separação do conjunto de treino e teste. Observa-se

que o *Gradient Boosting* obteve melhores resultados. Realizamos um teste-t para comparar os resultados, o que nos permite concluir que a diferença em relação aos outros métodos é estatisticamente significativa. As Tabelas 2 a 5 mostram as matrizes de confusão obtidas para as previsões realizadas por cada modelo, incluindo a **revocação** (fração de instâncias relevantes que são recuperadas) e a **precisão** (fração de instâncias recuperadas que são relevantes), indicando que todos os modelos sofrem de um mesmo problema: a maioria das instâncias de cada classe tende a ser classificada como proficiência intermediária. Além disso, existe um número relativamente grande de instâncias do tipo iniciante sendo classificadas como tipo avançado. No caso do KNN e Gradient Boosting, o problema oposto também acontece.

		Estimado			
		0	**1**	**2**	Revocação
Real	**0**	**555**	3154	265	14,0%
	1	302	**3119**	343	82,9%
	2	90	1069	**204**	19,3%
	Precisão	58,6%	42,5%	25,1%	

Tabela 2. Logistic Regression

		Estimado			
		0	**1**	**2**	Revocação
Real	**0**	**990**	2812	172	24,9%
	1	196	**3409**	159	90,6%
	2	88	948	**327**	24,0%
	Precisão	77,7%	47,6%	26,1%	

Tabela 3. Random Forest

		Estimado			
		0	**1**	**2**	Revocação
Real	**0**	**904**	2720	350	22,8%
	1	547	**2790**	427	74,1%
	2	188	931	**244**	17,9%
	Precisão	55,2%	43,3%	23,9%	

Tabela 4. KNN

		Estimado			
		0	**1**	**2**	Acurácia
Real	**0**	**1845**	1885	244	46,4%
	1	570	**2903**	291	77,1%
	2	217	720	**426**	31,3%
	Precisão	70,1%	52,7%	44,3%	

Tabela 5. Gradient Boosting

6.2. Resultados de classificação usando SEMPLICe

Os resultados obtidos através dos vários modelos de classificação demonstram que não é trivial estimar o nível de proficiência de um usuário do Reddit a partir das características textuais de uma única publicação (p. ex., auenticidade ou quão analítico é uma publicação). Esta observação motiva o uso de técnicas mais sofisticadas, como o SEMPLICe, que utiliza informações sobre a sequência de posts de um usuário.

Dentre os modelos investigados na abordagem não-sequencial, o Gradient Boosting é aquele que obteve o melhor resultado. Por esta razão, selecionamos este classificador para calcular as probabilidades que cada post tenha sido escrito por um usuário de nível de proficiência iniciante, intermediário ou avançado. Estas probabilidades servem como insumo para o SEMPLICe.

	Não-sequencial (GB)	GB+SEMPLICe
Média	0,4568	0,5919
Desvio	0,0093	0,0098

Tabela 6. Média e desvio padrão do F_1 ponderado obtido pelo Gradient Boosting (GB) e pelo SEMPLICe, usando as previsões retornadas pelo GB como insumo. Modelo proposto apresenta um ganho de 29.6% em relação ao melhor modelo não-sequencial.

Para facilitar a comparação das duas abordagens, repetimos na Tabela 6 o F_1 ponderado do Gradient Boosting (GB) e incluímos o respectivo resultado para o SEMPLICe quando as previsões do GB são usadas como insumo. Após o uso do algoritmo

SEMPLICe, o F_1 ponderado aumentou de 0,4568 para 0,5919, representando um ganho de 29.6% em relação ao melhor modelo não-sequencial em nossos experimentos. As publicações foram classificadas corretamente com uma precisão de 62,74%.

		Estimado			
		0	**1**	**2**	Revocação
Real	**0**	**1754**	2149	70	45,2%
	1	150	**3543**	70	94,2%
	2	63	976	**324**	23,8%
	Precisão	89,2%	53,1%	69,8%	

Tabela 7. Matriz de confusão do SEMPLICe usando Gradient Boosting como entrada. Há um aumento na revocação da classe intermediária e, apesar da redução na revocação da classe avançada, a ocorrência de erros grosseiros (classificar iniciante como avançado e vice-versa) cai drasticamente.

A Tabela 7 contém a matriz de confusão para os resultados do SEMPLICe alimentado pelo GB. Em relação aos resultados do GB, observa-se que o SEMPLICe causou um aumento na revocação da classe intermediária de 77% para 94%, enquanto a classe iniciante teve uma pequena queda de 46% para 45%, e a classe avançada teve uma queda maior, descendo de 31% para 24%. No entanto, o número de erros grosseiros (classificar iniciante como avançado e vice-versa) reduziu de maneira drástica. Discutimos este resultado em detalhe na seção seguinte.

6.3. Discussão e Limitações dos Dados

Os resultados obtidos pelo SEMPLICe demonstraram um ganho de 29,7% na F_1 ponderada em relação às previsões originais do Gradient Boosting (GB), usadas como entrada. De maneira geral, o SEMPLICe obteve uma revocação bastante elevada para categoria intermediário, teve uma revocação maior ou igual a dos outros modelos para categoria iniciante e teve uma revocação menor que o GB para a categoria avançada, embora tenha classificado menos instâncias desta classe como iniciantes (erros grosseiros).

Para melhor entender a composição do erro das previsões retornadas pelo SEMPLICe, agrupamos os usuários conforme suas curvas de aprendizado, por exemplo, um usuário do tipo $1 \rightarrow 2$ é um usuário que em algum momento foi de iniciante para intermediário. A Tabela 8 mostra a taxa de acerto médio por grupo. Observa-se que a maioria das publicações associadas à proficiência avançada (i.e., índice 3) correspondem a usuários que já entram na comunidade com tal nível. A taxa de acerto média para estas publicações é baixa (23,95%). Uma possível explicação para este fato é que a comunidade em estudo é voltada para o compartilhamento de informações entre usuários dos mais diversos níveis e, dessa maneira, usuários avançados podem buscar utilizar uma linguagem mais simples ao responder questões de usuários menos proficientes.

	Sequências de proficiências						
	1	2	3	$1 \rightarrow 2$	$1 \rightarrow 3$	$2 \rightarrow 3$	$1 \rightarrow 2 \rightarrow 3$
Total de usuários	606	417	132	43	3	16	2
Média de posts	11	11	11	10	14	11	9
Taxa de acerto médio	47,19%	89,19%	23,95%	66,73%	20,00%	61,23%	72,36%

Tabela 8. Estatísticas do SEMPLICe para usuários agrupados por sequência de proficiência (1: iniciante, 2: intermediário, 3: avançado, $\rightarrow$: há transição).

Os dados utilizados também podem conter imprecisões, porque não é garantido que o usuário escreveu o próprio texto (e.g., um usuário iniciante pode publicar um texto mais complexo, pedindo ajuda para tradução). Além disso, não é possível afirmar que os níveis de proficiência indicados pelos usuários estão corretos, dado que a proficiência é auto-declarada. Para contornar essa limitação, uma possível abordagem seria identificar aqueles usuários que possuem alguma certificação indicada em seu perfil.

Outra limitação dos dados é que são apenas textos curtos e informais, escritos na Web. Apesar da quantidade de textos, muitos podem conter gírias ou abreviações comuns à Web, o que pode prejudicar essa análise. Outros trabalhos nessa linha utilizaram textos de testes de proficiência, em que voluntários respondem questões com a finalidade de medir a proficiência. Textos de redes sociais, em contrapartida, não são escritos com essa finalidade, o que pode impactar diretamente nos resultados da análise.

7. Conclusão

Neste trabalho, coletamos dados de uma comunidade para aprendizado de alemão no Reddit a fim de treinar modelos para classificação automática de proficiência. Isto foi possível porque uma grande quantidade das publicações (posts ou comentários) neste subreddit possui uma *tag* que funciona como uma auto-declaração do nível de proficiência do usuário no momento em que são escritas.

Realizamos a análise de *posts* e comentários de usuários da comunidade de alemão que indicaram, no momento da publicação, qual era a sua proficiência atual. Por meio da ferramenta LIWC extraímos características textuais das publicações que usamos para treinar classificadores tradicionalmente encontrados na literatura: Logistic Regression, Random Forest, KNN e Gradient Boosting (GB). Esta abordagem, que denominamos não-sequencial, não considera a identidade do usuário que escreveu as publicações, tampouco a ordem em que elas aparecem. Embora o GB tenha obtido os melhores resultados (F_1 ponderado igual a 0,46), ele ainda comete erros grosseiros, classificando muitos usuários iniciantes como avançados e vice-versa.

Como principal contribuição deste trabalho, propomos um método que permite melhorar as previsões feitas por um classificador de proficiência para publicações individuais ao agrupá-las por usuário e ordená-las no tempo. Este método assume que o nível de proficiência é uma função monotonicamente não-decrescente. Apresentamos um algoritmo força bruta para este método cuja complexidade é $\Theta(N^2)$ onde N é o número de publicações escritas por um usuário. Propomos em seguida um algoritmo baseado no paradigma de programação dinâmica que retorna os mesmos resultados, mas com complexidade $\Theta(N)$. Denominamos este algoritmo SEMPLICe.

Através do SEMPLICe foi possível aprimorar as previsões, elevando o F_1 ponderado em 29,6%. A acurácia, que ultrapassa 62%, é um valor próximo dos 70% encontrados por outras pesquisas que realizaram classificação automática de proficiência a partir de textos mais longos e formais [Yang et al. 2016]. O SEMPLICe também reduziu de forma substancial a ocorrência de erros grosseiros. A maioria dos erros cometidos pelo SEMPLICe são nas classificações de publicações escritas por usuários que já entram na comunidade como usuários avançados. Uma das hipóteses que explicaria este fenômeno e que pretendemos investigar futuramente é a de que tais usuários buscam utilizar uma linguagem mais simples ao interagir com usuários menos proficientes.

Os resultados deste trabalho podem ser utilizados por usuários de mídias sociais, voltadas ou não para aprendizado de idiomas, para acompanhar a evolução de proficiência ao longo do tempo. Por exemplo, podem ser utilizados para auxiliar no direcionamento das atividades de uma comunidade de aprendizado de língua, baseado na proficiência de seus usuários. Trabalhos futuros incluem a realização de experimentos com outros idiomas, como espanhol e francês, assim como utilizar dados de outras redes além do Reddit. Adicionalmente, almeja-se a criação de um *bot* do *Reddit* que identifica quando um usuário provavelmente melhorou seu nível de proficiência e o notifica sobre isso.

Referências

Altman, N. S. (1992). An introduction to kernel and nearest-neighbor nonparametric regression. *The American Statistician*, 46(3):175–185.

Arnold, N. and Paulus, T. (2010). Using a social networking site for experiential learning: Appropriating, lurking, modeling and community building. *The Internet and higher education*, 13(4):188–196.

Bergstra, J. and Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13(Feb):281–305.

Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5–32.

Crossley, S. A., Salsbury, T., and McNamara, D. S. (2012). Predicting the proficiency level of language learners using lexical indices. *Language Testing*, 29(2):243–263.

Friedman, J. H. (2002). Stochastic gradient boosting. *Computational Statistics & Data Analysis*, 38(4):367–378.

Levy, M. (1997). *Computer-assisted language learning: Context and conceptualization*. Oxford University Press.

Lin, C.-H., Warschauer, M., and Blake, R. (2016). Language learning through social networks: Perceptions and reality.

Spolaôr, N. and Tsoumakas, G. (2013). Evaluating feature selection methods for multi-label text classification. *BioASQ workhsop*.

Warschauer, M. and Healey, D. (1998). Computers and language learning: An overview. *Language teaching*, 31(2):57–71.

Yang, Y., Yu, W., and Lim, H. (2016). Predicting second language proficiency level using linguistic cognitive task and machine learning techniques. *Wireless Personal Communications*, 86(1):271–285.

Yu, H.-F., Huang, F.-L., and Lin, C.-J. (2011). Dual coordinate descent methods for logistic regression and maximum entropy models. *Machine Learning*, 85(1-2):41–75.

Zhao, Y. (1996). Language learning on the world wide web: Toward a framework of network based call. *Calico Journal*, pages 37–51.

Zourou, K. (2012). De l'attrait des médias sociaux pour l'apprentissage des langues–regard sur l'état de l'art. *Alsic. Apprentissage des Langues et Systèmes d'Information et de Communication*, 15(1).