Avaliação automática de respostas curtas através de uma arquitetura de Deep Learning

Leonardo Comelli
ICMC-USP, São Carlos
São Paulo, Brasil
leonardo.comelli@gmail.com

***Resumo*—Com o aumento dos cursos no formato EAD, a avaliação automática de respostas curtas se tornou uma necessidade, visando poupar tempo do professor e diminuindo o tempo de *feedback* para o aluno. Este artigo apresenta, em linhas gerais, alguns artigos que já propuseram uma solução para esta tarefa, além de executar uma prova de conceito utilizando *word embeddings* previamente treinadas contra o *corpus* disponibilizado no *SemEval 2013 - Task 7*. O melhor resultado obtido foi uma acurácia de 82,187%.**

I. Introdução

A avaliação de ensino de um aluno esta presente em quase todo o seu período da escolarização, sendo que, em grande parte, esta avaliação é realizada através de questões com respostas curtas. Este modelo de avaliação tem grande relevância, pois avalia a capacidade de leitura, interpretação e criação do aluno. Contudo, o processo de correção manual das respostas exige muito tempo e atenção do professor, além de retardar o *feedback* ao aluno[1].

Com a modernização do sistema de ensino e a crescente oferta de cursos de ensino a distância, que no ano de 2015 ultrapassou a marca dos 2 milhões de alunos[2], a pesquisa por soluções de avaliação automática de respostas esta, cada vez mais, sendo explorada.

Entretanto, o foco em novas soluções para avaliação automática de respostas não deve-se apenas ao crescimento de cursos no formato EAD (Ensino a Distância), a nova geração de processadores (GPU - Graphics Processing Unit) e a utilização de redes neurais multi-camadas (Deep Learning) tem exposto uma série de possibilidades que ainda não haviam sido exploradas.

No sentido de contribuição, este artigo tem a pretensão de apresentar uma breve revisão sobre o tema e fornecer uma proposta de solução utilizando uma arquitetura de Deep Learning. Além disso, desenvolver uma primeira prova de conceito visando a avaliação automática de respostas curtas de um domínio específico e na lingua inglesa.

II. Trabalhos Relacionados

Em meados de 2003, a quantidade de avaliações de ensino e testes preparatórios realizados através da Internet aumentou consideravelmente e a correção automática passou a ser uma necessidade. Nesta época, o foco principal era a avaliação automatizada de redações (TCT[3], SEAR[4], Intelligent Essay Assessor[5], IEMS[6], etc), desprezando as questões com respostas curtas que, normalmente, estavam vinculadas aos deveres escolar, avaliações em sala de aula e revisões de capítulos[7].

Aproveitando essa lacuna e entendendo a necessidade de avaliar automaticamente as questões com respostas curtas a Educational Testing Service(ETS) Technologies, empresa especializada em aplicação e correção de testes para diversas instituições, desenvolveu o c-Rater. Uma ferramenta específica para respostas curtas que difere, em muitos pontos, das ferramentas existentes (na época) focadas em avaliação automatizada de redações[7].

Ainda em 2003, o UCLES Group (Cambridge-ESOL, Cambridge International Examinations e Oxford, Cambridge and RSA Examinations) que provê serviços de avaliação ao redor do mundo iniciou o desenvolvimento do OXFORD-UCLES, que utilizava técnicas similares ao c-Rater para garantir um processamento eficiente frente a dados com ruídos (erros gramaticais ou ortográficos). As técnicas de extração de informações baseadas em padrões foram amplamente utilizadas contra 201 questões de biologia retiradas do General Certificate of Secondary Education, obtendo a acurácia de 88% [8].

Passado algum tempo, a ETS Technologies publicou algumas evoluções no c-Rater, como a utilização de conceitos: onde, dado um conceito C e uma resposta de aluno A, ele verifica se C é uma inferência ou esta parafraseando A (em outras palavras, se A implica C então A é verdadeiro)[9]. O experimento realizado neste artigo, utilizou 12 questões de respostas curtas em inglês (7 compreensão de texto e 5 matemática) retiradas de um exame aplicado em alunos da 7ª e 8ª série. As questões foram previamente corrigidas e pontuadas por dois humanos, sendo que o melhor resultado das médias entre o c-Rater e cada humano (c-H1 / H2) foi 98%.

Embora a língua mais utilizado nas pesquisas seja a inglesa, existem alguns artigos publicados voltados para outras linguas, como por exemplo a língua portuguesa, que foi utilizada em uma pesquisa para avaliação automática de respostas discursivas utilizando um método LSA (Latent Semantic Analysis)[9]. Os testes para avaliar a eficiência desta proposta foram realizados a partir de amostras do corpus fornecido pela Universidade Federal do Pará, sendo: 130 questões de Biologia e 229 de Geografia. Visando garantir os melhores resultados, foi utilizado um modelo de regressão linear múltipla para cada disciplina, obtendo a acurácia de 85,62% para Biologia e

87,35% para Geografia.

III. Proposta

Esta pesquisa, tem como objetivo principal, verificar a eficiência da utilização de uma arquitetura Deep Learning na correção automática de respostas curtas. O fluxo da arquitetura proposta contém algumas etapas de pré-processamento (corretores), onde o foco será a eliminação de ruídos e uma manipulação eficiente dos dados, considerando que os dados fornecidos por alunos são, na maioria das vezes, constituídos por erros de ortografía e gramática. Este fluxo está representado na Figura 1.

Neste momento, o inutito é validar a acurácia da classificação textual baseada em *word embeddings* previamente treinadas. Por este motivo, o *corpus* utilizado para a validação foi estruturado e manipulado de antemão em outra pesquisa. O corpus em questão é o *SemEval 2013 - Task 7*[10] que será detalhado posteriormente.

A. Arquitetura

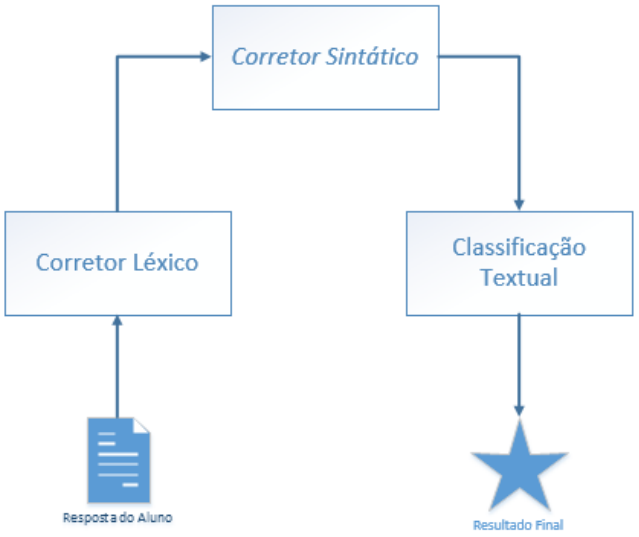

Figura 1. Arquitetura: Fluxo proposto

A entrada para o fluxo é a resposta do aluno que, primeiramente, deve passar pela etapa de correção léxica para detectar os erros de grafia. Para auxiliar na execução dessa etapa, uma sugestão de ferramenta é o Stanford Parser [11], que foi desenvolvimento dentro do *The Stanford Natural Language Processing Group* e pode ser utilizado com a língua portuguesa, além de possuir suporte a diversas linguagens de programação.

Na etapa seguinte, a correção sintática é executada para detectar os erros em relações entre as palavras. Para essa etapa a sugestão é a ferramenta CoGrOO [12], que é um corretor gramatical acoplável ao *LibreOffice* desenvolvido pelo Centro de Competência em Software Livre do IME/USP.

As ferramentas citadas para auxiliar nas etapas de correção possuem uma série de implementações prontas, porém é evidente que será necessário algum tipo de evolução para garantir uma maior eficiência dentro no domínio pesquisado. Os corretores (léxico e sintático) são importantes para diminuir os ruídos para as etapas subsequentes.

A última etapa da arquitetura é a principal, onde o esforço em validar a proposta foi concentrado. O objetivo dessa etapa é garantir que uma resposta correta de um aluno seja similar a resposta referência e uma resposta incorreta não tenha esta similaridade, baseado na definição de *Textual Entailment* onde: um texto T implica em outro texto H, se H é tido como verdade em todas as circunstâncias nas quais T é verdade [13].

Para esta etapa é necessário uma ferramenta que consiga medir a similaridade entre a resposta do aluno e uma ou mais respostas-referência. A opção escolhida para a prova de conceito foi o Doc2Vec [14] que utiliza *word embeddings* treinadas previamente contra as respostas-referência e/ou corretas para realizar a classificação textual.

O resultado esperado ao término da execução do fluxo é o quão uma resposta de um aluno é similar a uma respota referência, esta similaridade é representada por uma porcentagem.

B. Corpus

O *corpus* utilizado para a validação da proposta foi o disponibilizado no *SemEval 2013 - Task 7*[10]. Este corpus, na sua totalidade, é composto por 56 questões sobre eletrecidade/eletrônica com 3000 respostas e 197 questões de 15 domínios variados com 10000 respostas, todas as perguntas e respostas estão na língua inglesa. As respostas estão classificadas em: *5-way* (correta, parcialmente correta, contraditória, irrelevante e fora do domínio), *3-way* (correta, contraditória e incorreta) e *2-way* (correta e incorreta).

```
<question qtype="Q_EXPLAIN_SPECIFIC" id="BULB_C_VOLTAGE_EXPLAIN_WHY1" module="FaultFinding" stype="EVALUATE">

  <questionText>Explain why you got a voltage reading of 1.5 for terminal 1 and the positive terminal.</questi

  <referenceAnswers>
    <referenceAnswer category="BEST" id="answer204" fileID="BULB_C_VOLTAGE_EXPLAIN_WHY1_ANS1">
        Terminal 1 and the positive terminal are separated by the gap
    </referenceAnswer>
  </referenceAnswers>

  <studentAnswers>
    <studentAnswer count="1" answerMatch="answer204" accuracy="correct" id="FaultFinding-BULB_C_VOLTAGE_EXPLAI
        positive battery terminal is separated by a gap from terminal 1
    </studentAnswer>
    <studentAnswer count="1" accuracy="incorrect" id="FaultFinding-BULB_C_VOLTAGE_EXPLAIN_WHY1.sbj8-l1.qa223">
        Because terminal 1 is connected to the positive battery terminal
    </studentAnswer>
  </studentAnswers>

</question>
```

Figura 2. Corpus: SemEval 2013 - Task 7 - Formato XML

Para a validação da prova de conceito foi retirado uma amostra das questões sobre eletrecidade/eletrônica, sendo: 1 questão com 151 respostas classificadas no *2-way* (correta e incorreta). Para facilitar a execução, a amostra das questões que serão utilizadas foram convertidas do formato XML (conforme Figura 2) para dois arquivos no formato texto, onde um contêm apenas as questões corretas e o outro as incorretas (apresentado na Figura 3).

O corpus completo do *SemEval 2013 - Task 7* esta disponível na internet de forma pública [15], assim como a amostra retirada para a validação da proposta [16].

```
terminal 1 and the positive terminal are separated by the gap
terminal 1 and the positive terminal are not connected
terminal 1 is connected to the negative battery terminal
terminal 1 is not separated from the negative battery terminal
terminal 1 and the positive battery terminal are in different electrical states
there is a gap between terminal 1 and the positive terminal
terminal 1 is not connected to the positive battery terminal
there was a gap between terminal 1 and the positive terminal
because terminal 1 is connected to the negative battery terminal
because terminal 1 is connected to the negative battery terminal
because terminal 1 was connected to the negative terminal of the battery
terminal 1 and the positive terminal are separated by the gap
terminal 1 and the positive terminal are not connected
terminal 1 is connected to the negative battery terminal
```

Figura 3. Corpus: SemEval 2013 - Task 7 - Formato TXT

C. Implementação

A prova de conceito foi implementada com o apoio de alguns bibliotecas *python* que auxiliam o desenvolvimento de soluções de Processamento de Linguagem Natural, com destaque para o *gensim* [17] que possui uma implementação do Doc2Vec e o *scikit-learn* [18] que disponibiliza uma série de algoritmos para análise dados.

```
model = Doc2Vec(min_count=1, window=1, size=DOC2VEC_SIZE,
                negative=5, workers=7)
model.build_vocab(sentences.to_array())

for epoch in range(15):
    model.train(sentences.sentences_perm())

model.save('./sfta.d2v')
model = Doc2Vec.load('./sfta.d2v')

...

classifier = LogisticRegression()
classifier.fit(train_arrays, train_labels)

print classifier.score(test_arrays, test_labels)
```

Figura 4. Implementação: Trecho do código

Os exemplos disponibilizados no site do *gensim* foram a base para o desenvolvimento, conforme trecho de código apresentado na Figura 4. A maior parte do *corpus* foi utilizada para o treino e o restante para a realização dos testes. Buscando assegurar uma maior qualidade nos resultados obtidos, foram executados diversos testes sobre o mesmo corpus, sempre construindo os fragmentos de treino e teste de forma aleatória.

Ao final de cada execução, com o suporte da biblioteca *scikit-learn*, uma regressão logística é calculada com o intuito de tornar os resultados mais apurados.

Todos os artefatos desenvolvidos foram disponibilizados publicamente na internet [16].

IV. Resultados

O resultado foi obtido diante de um *corpus* com 151 respostas únicas, sendo que 135 foram utilizadas para treino e 16 para os testes. Para aumentar o tamanho do *corpus*, foram realizadas 10 iterações sobre o mesmo, separando os fragmentos de teste e treino de aleatória na mesma proporção citada anteriormente, totalizando 1600 testes.

A primeira validação da proposta atingiu a acurácia de 82%, conforme mostrado na Tabela I. Embora o resultado esteja abaixo de outras pesquisas, é importante ressaltar a arquitetura enxuta e o *corpus* com poucos ruídos.

Tabela I
Resultados - Acurácia: Testes com Doc2Vec

Testes	Acertos	Acurácia %
160	132	82,500
160	135	84,375
160	130	81,250
160	129	80,625
160	130	81,250
160	130	81,250
160	126	78,750
160	138	86,250
160	133	83,125
160	132	82,500
1600	**1315**	**82,187**

Comparando os resultados com outras pesquisa (vide Figura 5), fica evidente a proximidade dos resultados e possibilidade de melhorar, significativamente, os resultados obtidos na primeira prova de conceito no decorrer da pesquisa.

Pesquisa Realizada	Acurácia
c-rater: Automatic Content Scoring for Short Constructed Responses [7]	98%
auto-marking: using computational linguistics to score short, free text responses [8]	88%
Aplicação de um método LSA na avaliação automática de respostas discursivas [1]	87,35%

Figura 5. Resultados: Comparação com outras pesquisas

V. Conclusão e Trabalhos Futuros

O foco deste artigo foi a avaliação automática de respostas curtas, demonstrando de forma prática a possibilidade de validar a similaridade entre a resposta do aluno e uma ou mais respostas-referência utilizando técnicas de *word embeddings* através do Doc2Vec. Durante a pesquisa, notou-se que a similaridade entre respostas curtas pode ser muito sensível dependendo do domínio e os resultados podem, em alguns casos, conter "falso-verdadeiro".

Os resultados obtidos não estão acima do estado da arte para a tarefa proposta, porém são satisfatórios e importantes para a evolução da arquitetura apresentada.

VI. Trabalhos Futuros

Os próximos passos serão a execução de testes utilizando o *corpus* da plataforma de ensino chamado Alura [19], onde o domínio principal é a área de tecnologia e programação. Esta plataforma possui 232 cursos, 1734 aulas e 8170 questões na lingua portuguesa; sendo que já foi liberado para testes iniciais uma amostra com 06 questões e mais de 7000 respostas. Provavelmente a mudança de idioma implicará em alguns ajustes nas etapas previstas e, até mesmo, a inclusão de novas.

Além do trabalho com outro *corpus*, esta previsto o estudo de redes neurais multi-camadas, como por exemplo: LSTM [20] e/ou CNN [21], em conjunto com as *word embbedings*.

Como objetivo secundário, todos os artefatos serão sempre disponibilizados como código aberto para a comunidade, exceto as informações disponibilizados por terceiros em caráter confidencial.

AGRADECIMENTOS

Em especial à professora Dra. Sandra Aluísio do curso Tópicos em Inteligência Artificial do ICMC/USP e a todos os colegas da turma por compartilhar o conhecimento que foi vital para a confecção deste artigo.

REFERÊNCIAS

[1] João Carlos Alves dos Santos, Tácio Ribeiro, Eloi Favero, Joaquim Queiroz, [2012], *"Aplicação de um método LSA na avaliação automática de respostas discursivas."*, Anais do Workshop de Desafios da Computação Aplicada à Educação.

[2] ABED – Associação Brasileira de Educação a Distância, *"Censo EAD Brasil 2014, 2015"*. Disponível em: http://www.abed.org.br/censoead2014, acessado em: junho de 2016.

[3] Larkey, L., [1998], *"Automatic essay grading using text categorization techniques."*, In In proceedings of the 21st annual international ACM SIGIR Conference on Research and Development in Information Retrieval.

[4] Christie, J. [1999], *"Automated essay marking for both content and style."*, In Proceedings of the 3rd International Computer Assisted Assessment Conference.

[5] Foltz, P.; Laham, D.; and Landauer, T. [2003], *"Automated essay scoring."*, In Applications to educational technology.

[6] Ming, Y.; Mikhailov, A.; and Kuan, T. L. [2000], *"Intelligent essay marking system."*, Technical report, Learner Together NgeANN Polytechnic, Singapore.

[7] Leacock, C., and Chodorow, M., [2003], *"C-rater: Automated scoring of short-answer questions."*, Computers and Humanities, 389–405.

[8] Sukkarieh, J. Z.; Pulman, S. G.; and Raikes, N., [2003], *"Auto-marking: using computational linguistics to score short, free text responses."*, In Presented at the 29th IAEA.

[9] Sukkarieh, Jana Zuheir, and John Blackmore, [2009], *"c-rater: Automatic Content Scoring for Short Constructed Responses."*, FLAIRS Conference.

[10] Dzikovska, M. O., Nielsen, R. D., Brew, C., Leacock, C., Giampiccolo, D., Bentivogli, L., ... & Dang, H. T, [2013], *"SemEval-2013 task 7: The joint student response analysis and 8th recognizing textual entailment challenge."*, NORTH TEXAS STATE UNIV DENTON.

[11] The Stanford Natural Language Processing Group, *"The Stanford Parser: A statistical parser"*, Stanford University. 2005. Disponível em: http://nlp.stanford.edu/software/lex-parser.html, acessado em: agosto de 2016.

[12] CCSL - Centro de Competência em Software Livre, *"CoGrOO – Corretor Gramatical acoplável ao LibreOffice"*, IME - USP. 2013. Disponível em: http://cogroo.sourceforge.net, acessado em: junho de 2016.

[13] Dagan, I., Glickman, O. e Magnini, B., [2006], *"The PASCAL Recognising Textual Entailment Challenge"*, Proceedings of the First International Conference on Machine Learning Challenges: Evaluating Predictive Uncertainty Visual Object Classifi- cation, and Recognizing Textual Entailment.

[14] Quoc V. Le, Tomas Mikolov, *"Distributed Representations of Sentences and Documents"*, 2016. Disponível em: http://arxiv.org/abs/1405.4053, acessado em: agosto de 2016.

[15] Dzikovska, M. O., Nielsen, R. D., Brew, C., Leacock, C., Giampiccolo, D., Bentivogli, L., ... & Dang, H. T, *"SemEval-2013 task 7: The joint student response analysis and 8th recognizing textual entailment challenge."*, 2013. Disponível em: https://www.cs.york.ac.uk/semeval-2013/task7/index.php%3Fid=data.html, acessado em: agosto de 2016.

[16] Comelli, Leonardo, *"Amostra do Corpus SemEval-2013 task 7"*, 2016. Disponível em: https://github.com/leocomelli/sfta, acessado em: agosto de 2016.

[17] RaRe Technologies, *"Topic Modelling for Humans"*, 2016. Disponível em: http://radimrehurek.com/gensim, acessado em: agosto de 2016.

[18] scikit-learn, *"scikit-learn: Machine Learning in Python"*, 2016. Disponível em: http://scikit-learn.org, acessado em: agosto de 2016.

[19] Alura, *"Cursos online de tecnologia que reinventam sua carreira"*, 2016. Disponível em: https://www.alura.com.br/, acessado em: agosto de 2016.

[20] SkyMind, *"A Beginner's Guide to Recurrent Networks and LSTMs"*, 2016. Disponível em: http://deeplearning4j.org/lstm, acessado em: agosto de 2016.

[21] SkyMind, *"Convolutional Networks in Java"*, 2016. Disponível em: http://deeplearning4j.org/convolutionalnets, acessado em: agosto de 2016.