

MINISTÉRIO DA DEFESA
EXÉRCITO BRASILEIRO
DEPARTAMENTO DE CIÊNCIA E TECNOLOGIA
INSTITUTO MILITAR DE ENGENHARIA

Victor Villas Bôas Chaves
Lucas Sousa Meireles
Claudio Cavalcante Bomfim Júnior

Identification by Keystroke Dynamics

Rio de Janeiro
Dezembro 2017

Instituto Militar de Engenharia

Victor Villas Bôas Chaves

Identification by Keystroke Dynamics

Relatório Final do Programa Institucional de Bolsas de
Iniciação em Desenvolvimento Tecnológico e Inovação
do CNPq / Instituto Militar de Engenharia.
Orientador: Prof. Ronaldo Goldschmidt - D.C.

Rio de Janeiro
Dezembro 2017

cDezembro 2017

INSTITUTO MILITAR DE ENGENHARIA
Praça General Tibúrcio, 80 - Praia Vermelha
Rio de Janeiro-RJ CEP 22290-270

Este exemplar é de propriedade do Instituto Militar de Engenharia, que poderá incluí-lo em base de dados, armazenar em computador, microfilmар ou adotar qualquer forma de arquivamento.

É permitida a menção, reprodução parcial ou integral e a transmissão entre bibliotecas deste trabalho, sem modificação de seu texto, em qualquer meio que esteja ou venha a ser fixado, para pesquisa acadêmica, comentários e citações, desde que sem finalidade comercial e que seja feita a referência bibliográfica completa.

Os conceitos expressos neste trabalho são de responsabilidade do(s) autor(es) e do(s) orientador(es).

/ Victor Villas Bôas Chaves; orientados por Prof. Ronaldo Goldschmidt- Rio de Janeiro: Instituto Militar de Engenharia, Dezembro 2017.

15p. : il.

- Instituto Militar de Engenharia - Rio de Janeiro, Dezembro 2017.

Resumo

Nosso resumo **Palavras-chaves:** dinâmica da digitação, reconhecimento de usuário

Abstract

Our big abstract **Keywords:** our keywords

Sumário

1	INTRODUÇÃO	7
1.1	Contexto e Motivação	7
1.2	Objetivos	7
1.3	Contribuições Esperadas	7
1.4	Método	7
1.5	Cronograma	7
1.6	Viabilidade	7
1.7	Organização do Texto	8
2	FUNDAMENTAÇÃO	9
2.1	Problema de Classificação	9
2.2	Erro de amostragem	9
2.3	Dimensão de Vapnik-Charvanenks (D_{vc})	11
2.4	Validação	11
2.5	Trabalhos Relacionados	12
3	SOLUÇÃO PROPOSTA	13
3.1	Modelo conceitual	13
3.2	Protótipo	13
4	EXPERIMENTOS E RESULTADOS	14
5	CONCLUSÃO	15
	REFERÊNCIAS	16

1 Introdução

1.1 Contexto e Motivação

Sistemas de Segurança da Informação modernos não se baseiam em um único método de autenticação, mas incrementalmente adicionam mecanismos com múltiplos fatores. Quanto mais e melhores fatores, maior a certeza da identidade ser autenticada corretamente.

Dentre as alternativas mais promissoras estão os fatores biométricos, valorizados por sua natureza individual e difícil falsificação. Os fatores biométricos frequentemente mencionados são os fisiológicos, mas seu emprego traz diversos fatores complicantes como a necessidade de amostragem prévia e diminuição da usabilidade do sistema. Uma alternativa é o uso de fatores biométricos de comportamento, como padrões comportamentais expressos naturalmente pelo usuário.

As vantagens dos fatores comportamentais incluem a possibilidade de amostragem silenciosa, maior variabilidade do grau de confiança e a transparência do mecanismo para o usuário. Em particular, sistemas providos pela *Web* em geral possuem uma uniformidade de interface que permite a coleta de vários padrões comportamentais durante todo o uso do sistema.

Alguns exemplos de comportamentos de interesse coletáveis incluem padrões de digitação, cliques de *mouse* ou áreas do sistema e recursos acessadas pelo usuário. A pesquisa sobre como utilizar fatores dessa natureza pode impulsionar sistemas mais seguros e menos impactantes na experiência do usuário.

1.2 Objetivos

Neste trabalho serão investigados os processos necessários para se utilizar os padrões de digitação como fator de autenticação biométrica comportamental. Tais processos incluem a coleta de dados, extração de informação, algoritmos de decisão e arquiteturas de sistema que tornem possível a implantação deste método.

- Analisar os tipos de informação que se podem extrair a partir dos padrões de digitação de um indivíduo;
- Modelar a combinação das informações extraídas utilizando algoritmos de aprendizado de máquina;
- Sistematizar um mecanismo de coleta de amostras que permita o treinamento dos modelos escolhidos;
- Definir uma arquitetura de sistema para implantação dos mecanismos de coleta e autenticação definidos;

1.3 Contribuições Esperadas

1.4 Método

1.5 Cronograma

1.6 Viabilidade

O que eu escrevo aqui?

	Fev	Mar	Abr	Mai	Jun	Jul	Ago	Set	Out
Revisão bibliográfica									
Modelagem conceitual									
Prototipagem									
Coleta de dados									
Entrega de relatório parcial									
Experimentação									
Entrega do relatório final									
Apresentação									

Tabela 1 – Cronograma mensal de trabalho

1.7 Organização do Texto

No capítulo 2.4 são introduzidos os conceitos necessários para a modelagem conceitual de um sistema de autenticação por dinâmica de digitação. Na seção 2.5 são discutidos trabalhos relacionados.

No capítulo 3 é apresentado uma arquitetura de sistema de autenticação isolado, para fácil implantação do método apresentado. Na seção 3.1 é definido o modelo de autenticação, especificando o fluxo de informações desde a coleta até a decisão de um grau de confiança de identidade, enquanto em 3.2 é demonstrada uma possível implementação da solução proposta, servindo como prova de conceito para o modelo.

No capítulo 4 são analisados os resultados do experimento proposto com o protótipo criado, analisando o sucesso da solução.

No capítulo 5 termina-se por sumarizar o conceito, a solução e os resultados obtidos pelo sistema apresentado.

2 Fundamentação

O problema que o ramo de aprendizado por máquinas se propõe a resolver é a busca pela aproximação funcional matemática de um problema real em um conjunto matemático, onde $F : C \rightarrow R$ é a função alvo desconhecida, então para resolver o problema é suposto uma função $G : H \rightarrow R$, onde G é a função estimada que pretende-se aproximar de F , $H \subset C$ é o conjunto das amostras que pretende-se expandir para o conjunto real expandido C , os valores de H e R são conhecidos; porém, para um problema não interpolável, não há um mapeamento conhecido de $F : C \rightarrow R$, seja por falta parâmetros de difícil análise, ou pelo caráter indeterminado do problema, em ambos os casos o tratamento é similar através de estimação e tratamento de erros na abordagem.

2.1 Problema de Classificação

O problema de traduzir um conjunto amostral H em conjuntos distintos com uma abordagem não interpolável é conhecido como o problema classificação, para tal pode se usar, no domínio discreto, a seguinte abordagem. Seja $f : x \rightarrow y$ a função alvo onde $x = [x_1, x_2, \dots, x_d]^t$ é o vetor das amostras de H , seja o somatório $\sum_{i=1}^d \omega_i * x_i$, onde ω_i é o peso do seu respectivo elemento do vetor x , de tal forma que:

$$\sum_{i=1}^d \omega_i * x_i > t; y = 1. \quad \sum_{i=1}^d \omega_i * x_i < t; y = 0. \quad (2.1)$$

De tal forma os pesos tem suas características definidas por seus respectivos elementos de x da seguinte forma:

$|\omega_i|$ é alto quando x_i for importante.

$\omega > 0$ quando x_i for benéfico.

$\omega < 0$ quando x_i for maléfico.

Por fim teremos uma aproximação da função alvo no problema de classificação para $h(x) = \text{sign}((\sum_{i=1}^d \omega_i * x_i) + \omega_0)$, onde $\omega_0 = t$.

Neste caso o problema de classificação se restringirá a apenas dois conjuntos alvo em R , para uma generalização com múltiplos conjuntos usam-se estratégias de modo a recair no problema com dois conjuntos alvo.

2.2 Erro de amostragem

Vale notar que o vetor x apenas representa as amostras colhidas, assim embora ele represente o conjunto C como uma aproximação de C para H existe uma divergência conhecida como erro dentro da amostra E_{in} essa diferença é representada pela diferença entre x e sua projeção em C , assim como a extrapolação da função nem sempre vai atender o comportamento esperado pela função estimada $G : H \rightarrow R$ assim a divergência entre o real e o esperado é o chamado fora da amostra E_{out} , o qual funciona como um meio de verificar o quão errada esta a estimação de $G : H \rightarrow R$.

Em um problema de aprendizado a verificação de erros não é um problema trivial, uma vez que pela impossibilidade de se conferir ambos os erros, E_{in} e E_{out} , de maneira direta, por não conhecermos a função

alvo, usam-se ferramentas que possibilitam verificar tais erros e assim modificar de maneira eficiente o método de estimação da função, uma primeira ferramenta usada para verificação de erros é a desigualdade de Hoeffding Chernoff como meio de verificação sobre os erros dentro e fora da amostra, a desigualdade em questão pode ser expressa da seguinte forma:

$$\begin{aligned} P[|v - \mu| > \epsilon] &\leq 2 * e^{-2 * \epsilon^2 * N}, \epsilon > 0 \\ P[|v - \mu| \leq \epsilon] &> 2 * e^{-2 * \epsilon^2 * N}, \epsilon > 0 \end{aligned} \quad (2.2)$$

A fim de melhor adaptar os valores, leia-se $P[x]$ como probabilidade de x , e diminuir os erros escolhe-se o valor de ϵ de tal forma que $v + \epsilon \geq \mu \geq v - \epsilon$, nota-se que a fronteira denotada por $2 * e^{-2 * \epsilon^2 * N}$ não depende de μ , nem do tamanho do domínio.

Adaptando-se a desigualdade ao problema de classificação ela é reescrita de forma que $|E_{in} - E_{out}| = |v - \mu|$ assim tem-se:

$$\begin{aligned} P[|E_{in} - E_{out}| > \epsilon] &\leq 2 * |H| * e^{-2 * \epsilon^2 * N} \\ P[|E_{in} - E_{out}| \leq \epsilon] &> 2 * |H| * e^{-2 * \epsilon^2 * N}, \epsilon > 0 \end{aligned} \quad (2.3)$$

Como o erro E_{in} é uma medida que advem da diferença entre a projeção da função estimada pode há uma identidade para tal, na qual o E_{in} é a média das distâncias entre o valor real e o estimado, os seja:

$$\begin{aligned} E_{in} &= \frac{\sum_{i=0}^N L(g(x_i), f(x_i))}{N} \\ E_{in} &= \frac{\sum_{i=0}^N L(g(x_i), y_i)}{N} \end{aligned} \quad (2.4)$$

Quando a função alvo é indeterminada, dessa forma como o E_{in} é independente de E_{out} além de ser identicamente distribuída a tal pode-se afirmar que o valor de E_{out} converge para $E_{out} = E(E_{in}) = E(\frac{\sum_{i=0}^N L(g(x_i), y_i)}{N})$, leia $E(x)$ como a esperança de x , logo o erro de classificação é representado como $P[g(x) \neq y] = E(L(g(x), y))$ de tal forma para minimizar de maneira mais eficiente se estipula parametros afim de diminuir o erro quadrático médio assim buscam-se os parâmetros, tais que $(x, y) \rightarrow \inf E((y - g(x))^2)$, leia-se \inf como infimo.

Afim de otimizar a estimação busca-se então um classificador ótimo ao problema tal que o E_{out} seja mínimo sem para isso alterar a amostra, de tal forma $E_{out}^* = \inf(E_{out}(h))$, $h \in H$ nesse caso o erro fora da amostra é minimizado quando ele se aproxima do erro para um classificador ótimo $\inf(E_{out} - E_{out}^*)$, seja $\bar{h} \in H$, tal que \bar{h} é a amostra ideal, nesse caso $E_{out} - E_{out}^* = (E_{out} - E_{out}(\bar{h})) + (E_{out}(\bar{h}) - E_{out}^*)$, onde $E_{out} - E_{out}(\bar{h})$ é o erro na estimação da amostra e $E_{out}(\bar{h}) - E_{out}^*$ o erro na aproximação da amostra, assim minimizar o erro de estimação implica em aumentar o erro de aproximação, bem como assim minimizar o erro de aproximação implica em aumentar o erro de estimação, no entanto o erro de estimação esta diretamente ligado ao E_{in} , portanto minimizar o E_{in} e consequentemente o E_{out} implicam em aumento do erro de aproximação, logo existe um risco associado quando se tenta adaptar a função estimada ao comportamento real, o que pode levar ao problema de overfitting, para impedi-lo avalia-se então o risco associado a relação entre os erros conforme aa seguinte forma:

$$E_{out} - E_{out}^* \leq E_{out} - E_{in} - E_{out}^* + E_{in}^* \leq \text{Sup}(|E_{out} - E_{in}|) \quad (2.5)$$

Assim usando-se da desigualdade (3) para isolar o valor de ϵ , tem-se que o valor de $Sup(|E_{out} - E_{in}|)$ substituindo o valor ϵ na equação resultante chega-se a:

$$Sup(|E_{out} - E_{in}|) \leq \alpha \sqrt{\frac{\log(|H|)}{N\delta}} \quad (2.6)$$

Dessa forma é possível limitar a variação máxima entre os dois erros afim de minimizar o risco associado ao overfitting, para usar essa relação de maneira formal no ambito do problema ainda é preciso analisar as caraterísticas do conjunto H , de onde são colhidas as amostras para teste e treino, para tal busca-se a divisão em subconjuntos de maneira que estes sejam disjuntos e H é então separado como um conjunto, tal que $H(s) = \{h(x_1), h(x_2), \dots, h(x_N)\}$, com $s \in X^N$, assim o crescimento de H será delimitado por 2^N e consequentemente o seu número de dicotomias ($m_h(H)$), que são as formas de separar H em duas partes, será a forma de se avaliar o crescimento de H afim de reduzir os riscos ao selecionar a abordagem e tratar os erros do problema de classificação.

2.3 Dimensão de Vapnik-Charvanenks (D_{vc})

Ainda assim é necessária uma ferramenta que defina qual o maior subconjunto de H , o qual possa ser separado dicotomicamente em H , pois a análise anterior foi feita integralmente com base no supremo da diferença entre os erros em H para tal foi criada a dimensão de Vapnik-Charvanenks, assim:

$$\begin{aligned} d_{vc}(H) &= \max\{N : m_H\} \\ d_{vc}(H) &= \infty, \text{ se } (m_h(N) = 2^N, \forall N) \end{aligned} \quad (2.7)$$

a dimensão VC em sí é uma ferramenta de difícil estimação, para facilitar a sua utilização em vista da minimização de risco é utilizado o Lema de Sauer, tomando $d_{vc} = d$, implicando em:

$$m_h(N) \leq \sum_{i=0}^d \binom{N}{i} = \phi_d(N) \leq \left(\frac{eN}{d}\right)^d \quad (2.8)$$

Ainda assim o ajuste ainda estará sujeito a erros de generalização quando a tese escolhida $G : H \rightarrow C$, quando for colocado em prática, tal erro é exposto como erro de generalização denotado como $argMin E_{in}(h) = arg_{h \in H} \frac{1}{N} \sum L(h(x), y)$, leia Min como mínimo, a fim de minimizar esses erros bem como seus riscos tornam-se necessárias táticas de verificação e validação da tese.

2.4 Validação

A validação é uma técnica que visa tornar a tática escolhida em otimal, assim sendo a validação é principalmente um meio de garantir que a abordagem escolhida para o problema esteja tão próxima da realidade quanto for possível, neste caso o conjunto de amostras é tratado para aprimorar a aplicação da tática, o tratamento pode funcionar de maneiras de ferentes de uma tática para outra, por fim a tática de validação mais utilizada e precisa quanto a validar o modelo é a validação cruzada, a qual apesar da eficiência e precisão pode não ser utilizada em detrimento de outras táticas por ser muito custosa em termos computacionais.

A validação cruzada se baseia em redistribuir a amostra de teste em diversas sub-amostras de treino para então escolher-se recursivamente tais conjuntos de amostras e usar o restante como amostra de treino esta abordagem visa estimar o valor do E_{out} que não pode ser calculado sem treino ou verificação do modelo já

em uso, então calculam-se os erros para assim se escolher como amostra de teste a amostra, a qual contenha os melhores índices de erro; porém, como visto previamente, forçar os valores de erro na amostra e fora dela para minimiza-los pode acarretar em problemas de overfitting e neste caso os erros calculados durante a validação iriam distuar drasticamente do seu valor real.

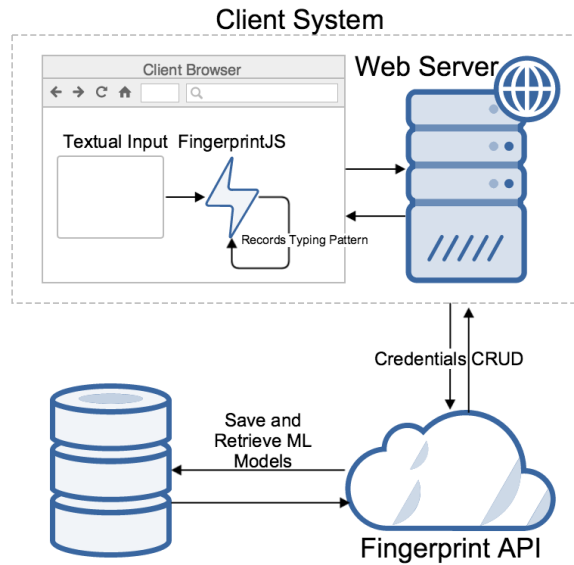
Outra técnica de validação amplamente utilizada é a reamostragem, nesse caso o modelo escolhido, mesmo após processos de verificação, não atende as expectativas quando implementado e por isso necessita-se refazer o treinamento, assim sendo se colhem novas amostras para refazer o processo de treino, é importante que não se usem as amostras já colhidas, pois estas iriam moldar o novo modelo com certo juízo de valor, contribuindo fatalmente para o problema de overfitting.

Outra forma de validar um modelo é pela verificação continua durante o treino, neste caso a cada iteração do treinamento calculam-se os erros do modelo em relação ao esperado para assim melhorar o modelo com alterações pontuais, similar ao processo de entrega de um produto em que a cada versão busca-se melhora-lo e consertar os problemas descobertos, assim como a tática de validação cruzada esse método é muito custoso computacionalmente e por isso pode ser desestimulado em relação aos outros.

2.5 Trabalhos Relacionados

3 Solução Proposta

Aqui entra a arquitetura do sistema, ver com Meireles.



3.1 Modelo conceitual

Aqui entra a modelagem do processo de decisão.

3.2 Protótipo

Aqui entra o nosso POC, sendo feito no GitHub.

4 Experimentos e Resultados

5 Conclusão

Excelente trabalho, time!

Referências