Implantação de um software detector de plágio para análise das questões dissertativas do ambiente virtual de aprendizagem TelEduc

Dra. Lucia R. H. R. Franco
lfranco@unifei.edu.br
Universidade Federal de Itajubá – UNIFEI
Itajubá - MG

M. Sc. José Renato Castro Milanez
jrcmilanez@gmail.com
Faculdade de Administração e Informática – FAI
Santa Rita do Sapucaí – MG

Flávia Aparecida Oliveira Santos
flaviaoliveira@unifei.edu.br
Universidade Federal de Itajubá – UNIFEI
Itajubá – MG

1 Resumo

Este artigo apresenta uma proposta de implantação de um software detector de plágio para análise das questões dissertativas do ambiente virtual de aprendizagem TelEduc. É apresentada ao leitor a definição de plágio e sua classificação quanto ao tipo. Adicionalmente, são apresentados alguns métodos utilizados para a descoberta de plágio em documentos eletrônicos. A fim de determinar o melhor software detector de plágio a ser utilizado, foram apresentados alguns softwares existentes, classificados pela sua distribuição e avaliados um a um com relação às suas funcionalidades. Dentre os softwares avaliados neste trabalho, o escolhido foi o Sherlock por apresentar a melhor relação custo-benefício baseada tanto por se tratar de um software livre quanto pela facilidade de integração com o TelEduc. A integração do Sherlock com o TelEduc foi implantada na Educação a Distância da Universidade Federal de Itajubá (UNIFEI) utilizando a linguagem de programação *PHP* (*PHP: Hypertext Preprocessor*). Os resultados apresentados foram obtidos após análise dos exercícios dissertativos aplicados aos alunos de um curso de especialização em Design Instrucional para EAD Virtual na modalidade a distância pelo projeto da Universidade Aberta do Brasil (UAB). Com base na bibliografia apresentada e nos resultados obtidos, concluiu-se que o software Sherlock possui funcionamento simples, uma vez que não possui técnicas avançadas na detecção de plágio. Tais técnicas são sugeridas para implementação em trabalhos futuros.

Palavras chave: Sherlock, TelEduc, plágio, detecção, educação, distância.

2 Abstract

This article presents an development proposal of software plagiarism detector for analysis of dissertative questions about virtual environment learning – TelEduc. It's presented to the reader the definition of plagiarism and its type classification. Additionally, some methods used for plagiarism discovery in electronic documents are presented. In order to determine the better software plagiarism detector to be used, some softwares were presented, classified for their distribution and evaluated one by one with regard to their functionalities. Among the softwares evaluated in this work, the chosen one was the Sherlock because it presents the best relation cost-benefit established as much for dealing with free software how much for the integration easiness with the TelEduc. The Sherlock integration with TelEduc was implanted at the Universidade Federal de Itajubá (UNIFEI) E-learning, using the programming language PHP (Hypertext Preprocessor). The presented results had been gotten after analysis of the applied dissertative exercises to the students of postgraduation course in Instructional Design for E-learning in distance modality for the Universidade Aberta do Brazil (UAB) project. On the bibliography basis presented and with the gotten results, it was concluded that Sherlock software has a simple functioning, however does not have advanced techniques in the plagiarism detention. Such techniques are suggested future works implementation.

3 Introdução

Os diversos meios tecnológicos existentes estão, cada vez mais, sendo utilizados por todos os segmentos educacionais facilitando o processo ensino-aprendizagem. Vários educadores têm se preocupado com a introdução do computador e softwares educacionais em sala de aula, procurando sistematizar o processo de aprendizagem para que os professores possam atuar com segurança junto aos educandos.

Os recursos tecnológicos disponíveis diminuem as dificuldades impostas pela distância física entre alunos e professores, possibilitando o armazenamento, distribuição e acesso às informações, independente do local. A tecnologia da informática permite criar ambientes virtuais de aprendizagem (AVA) nos quais alunos e professores sintam-se próximos, contribuindo para o aprendizado colaborativo.

Os fatores que contribuíram para o sucesso e a difusão da modalidade educação a distância (EaD) mediada pelo computador foram as facilidades proporcionadas pelo desenvolvimento tecnológico, que através da internet favoreceu a democratização do acesso à educação, flexibilidade e personalização da aprendizagem e incentivo da educação continuada. No entanto, propiciam também as oportunidades de plágio ou fraude, embora esses também ocorram na modalidade presencial. Portanto há uma grande preocupação dos profissionais da educação em diagnosticar o plágio nos meios eletrônicos.

O Ministério da Educação (MEC), através do decreto nº 6.303 de 12 de dezembro de 2007, estabelece que "As atividades presenciais obrigatórias, compreendendo avaliação, estágios, defesa de trabalhos ou prática em laboratório, conforme o art. 1º, § 1º, serão realizados na sede da instituição ou nos pólos de apoio presencial, devidamente credenciados.". Desta forma, as avaliações presenciais dos cursos a distância, a princípio efetivadas eletronicamente, dificultarão a incidência de plágio.

4 O plágio

Conforme Rosales et al. (2008) o plágio é o ato de assinar ou apresentar uma obra intelectual de qualquer natureza contendo partes de uma obra que pertença a outra sem que sejam dados os devidos créditos para o autor da obra consultada. Cada país trata o plágio com penalidades diferentes, mas independente do tratamento, a ética sempre deve prevalecer.

Liu et al. (2007) classificam dois tipos de plágio: o intra-corpal e o extra-corpal. O plágio intra-corpal é aquele no qual um sujeito copia a tarefa de outro quando ambos estão realizando uma mesma tarefa. Já o extra-corpal é aquele no qual o sujeito copia de fontes externas, como por exemplo, livro, artigo de revista, monografias ou internet.

4.1 Detectando o plágio em AVAs

Verificar o plágio nas tarefas submetidas nos AVAs esbarra na dificuldade de comparar as tarefas de todos os alunos. Manualmente, seria necessário comparar a tarefa de um aluno com a de todos os outros, o que para uma turma de 50 alunos torna-se extremamente demorada. Este problema faz com que os professores optem por tarefas nas quais os AVAs avaliam automaticamente os alunos como, por exemplo, alternativas de múltiplas escolhas, associação de colunas e verdadeiro ou falso, deixando de lado as questões dissertativas e a produção de textos.

A fim de aumentar as opções do professor, o plágio em questões dissertativas e produções textuais podem ser detectados através de softwares. De acordo com Maurer, Kappe e Zaka (2006), os métodos de descoberta de plágio através de softwares geralmente são divididos em três categorias: comparação entre documentos, busca por parágrafo suspeito na internet e a estilometria.

A comparação entre documentos é a mais comum. Nessa categoria, os documentos envolvidos são comparados entre si. Essa comparação pode ser feita de várias maneiras, de acordo com a implementação de cada software. Nos softwares mais simples, a comparação é feita de palavra em palavra. Já nos softwares mais complexos, a comparação é feita por parágrafos.

A busca por parágrafo suspeito na internet é geralmente implementada com o uso de ferramentas de busca como, por exemplo, o Google e o Yahoo!. O sucesso deste método só ocorrerá com textos publicados na internet e que estejam disponíveis sem custo algum para os usuários. Por exemplo, as buscas em artigos de revistas na maioria das vezes exigem que o usuário seja assinante da revista.

A estilometria analisa o estilo da escrita do texto através de comparações com documentos previamente escritos pelo mesmo autor. Este método é o mais complicado, pois envolve técnicas sofisticadas de inteligência artificial para a confecção do software. Porém, se o plágio for parafraseado, o estilo do autor original deixa de existir.

Neste trabalho, será adotada a comparação entre documentos das questões dissertativas do AVA TelEduc, detectando o plágio na forma intra-corpal.

4.2 Softwares detectores de plágio

Conforme o relatório desenvolvido por Scaife (2007) são apresentados alguns softwares detectores de plágio. Os softwares foram divididos em duas categorias em função da distribuição: comercial e livre.

4.2.1 Softwares comerciais

- **Copycatch:** utilizado para comparar documentos localmente disponíveis em banco de dados. Também oferece a versão on-line que estende as capacidades de detecção do plágio na Internet, usando a interface de programação de aplicativos (API) do Google.
- **Docoloc:** serviço on-line que oferece pesquisa, classificação e capacidade do Google API. O usuário do serviço envia o documento que precisa ser avaliado para um servidor que o analisa e envia um e-mail ao usuário com os fragmentos encontrados na internet.
- **Ephorus:** com mecanismo semelhante ao Docoloc.
- **Eve2 - Essay Verification Engine:** com mecanismo semelhante ao Docoloc.
- **GPSP - Glatt Plagiarism Screening Program:** armazena informações sobre o estilo da escrita de cada aluno. O autor de uma submissão suspeita tem que passar por um teste onde deve preencher uma palavra a cada cinco em todo o texto. O número de preenchimentos corretos e o tempo necessário para a conclusão do teste fornecem a hipótese de plágio.
- **MyDropBox:** conta com os mesmos recursos do Docoloc, porém utiliza arquivos de parceiros institucionais, todos protegidos por senha. O serviço usa busca proprietária e algoritmos estruturados que gera em média, relatórios em dois minutos. O software também é integrável com AVAs.
- **Turnitin:** concorrente do MyDropBox, contendo as mesmas funcionalidades.

4.2.2 Softwares livres

- **Copyscape:** com a mesma idéia central do Docoloc, porém totalmente gratuita.
- **DOC Cop:** realiza testes on-line utilizando identificação do cliente. O acesso é gratuito.
- **Plagiarism Checker:** ferramenta que simplesmente utiliza os buscadores *Google* ou *Yahoo!* para procurar frases desejadas. Esta ferramenta é extremamente simples e muito limitada.
- **Praise - Plotted Ring of Analysis of Similarity Exploration:** detecta a semelhança entre documentos. O resultado da análise pode ser visualizado pela ferramenta Vast para uma análise mais minuciosa do resultado.
- **Vast – Visualisation and Analisys of Similarity Tool:** software que proporciona de forma interativa a visualização entre

dois documentos. É utilizado em conjunto com a ferramenta Praise para a detecção e investigação de similaridade.

• **Urkund:** baseado em detecção on-line, oferece um serviço automatizado para detecção de plágio. Utiliza e-mail padrão para o sistema de submissão de documentos e visualização dos resultados.

• **WCopyfind:** detecta palavras ou frases de tamanho definido dentro de um repositório local de documentos.

• **Sherlock:** Encontra semelhanças entre documentos textuais, através de assinaturas digitais. Os textos devem estar armazenados em arquivos de texto puro e as assinaturas podem ou não ser armazenadas no disco rígido, a fim de acelerar comparações futuras. Também é faz parte da ferramenta BOSS, que é um sistema de submissão on-line de tarefas de estudantes de computação.

Entre os softwares apresentados, optou-se por usar o Sherlock devido à facilidade em encontrar seu código fonte e por possuir uma documentação sobre o seu funcionamento. Além destes benefícios, o Sherlock apresenta custo zero de aquisição por se tratar de um software livre.

4.3 O funcionamento do Sherlock

O software Sherlock (PIKE, 2008), conforme dito anteriormente, encontra semelhanças entre textos armazenados em arquivos do tipo texto puro. Para verificar a semelhança, o software analisa certa quantidade de palavras para cada linha do texto e gera uma assinatura digital que identifica essas palavras. Este procedimento de geração da assinatura digital é repetido até o final do documento.

Ao terminar esta etapa, o Sherlock possuirá as assinaturas digitais que identificam todo o texto. Para comparar texto com outro, o mesmo procedimento é realizado em outro texto a fim de se obter também as assinaturas digitais.

Finalmente, para determinar a semelhança entre os dois textos, o Sherlock compara as assinaturas digitais dos textos e retorna a porcentagem de semelhança entre eles. A comparação entre os textos realizada a seguinte maneira:

$$f1 = tamanhoDoArquivo1 = A + B$$

Equação 1

$$f2 = tamanhoDoArquivo2 = A + C$$

Equação 2

Onde A é a seção similar e B ou C são dissimilares. A similaridade é dada por:

$$Similaridade = 100 \times \frac{A}{(f1 + f2 - A)}$$

Equação 3

Porém, substituindo a Equação 1 e a Equação 2 na Equação 3, tem-se:

$$Similaridade = 100 \times \frac{A}{(A + B + A + C - A)}$$

$$Similaridade = 100 \times \frac{A}{(A + B + C)}$$

Equação 4

Na Equação 4, caso A, B e C sejam iguais, tem-se que a similaridade será 33%. Isto é desejável uma vez que o Sherlock determina a taxa de similaridade como uma fração da soma das similaridades com as dissimilaridades.

Outra informação importante sobre o funcionamento do Sherlock está na quantidade de comparações a serem realizadas para certa quantidade de textos. Uma vez que o Sherlock compara os textos em pares e todos os textos devem ser comparados entre si, a quantidade de comparações a serem realizadas será dada por:

$$C\binom{m}{2} = \frac{m!}{2! \times (m! - 2!)}$$

Equação 5

Onde m é a quantidade de textos a serem comparados. Desta forma, nota-se que é indesejável comparar um texto A com um texto B se B já foi comparado com A.

O software Sherlock possui os seguintes parâmetros a serem informados pelo usuário antes que seja realizada a comparação:

• **Zero bits (z):** controla a granularidade da comparação. Quanto maior o número, mais crua será a comparação, porém mais rápida. Quanto menor o número, mais exata a comparação, porém mais lenta e isso pode dificultar a detecção de plágio, pois pequenas mudanças no texto serão percebidas pelo software e não serão tratadas como semelhança.

• **Chain length (n):** controla quantas palavras formam uma assinatura digital. Isto também contribui para a granularidade da comparação. Quanto maior o número, maior a exatidão. Entretanto, a comparação será mais lenta.

- **Threshold (t):** controla o quanto similar devem ser os textos antes de serem processados.

O sucesso (ou fracasso) ao detectar o plágio com o Sherlock está intimamente ligado aos valores utilizados nestes parâmetros.

4.4 Teste de eficácia do Sherlock

Para verificar a eficácia do Sherlock, foi criada uma amostra baseada num texto contendo um total de 108 palavras. O texto foi então dividido entre 9 alunos, onde o texto do primeiro aluno terá as 12 primeiras palavras do texto da amostra, o segundo terá as primeiras 24 palavras e assim sucessivamente até que o nono aluno terá todas as 108 palavras em seu texto.

Desta forma, espera-se encontrar as seguintes porcentagens de plágio, entre o primeiro aluno e os outros oito alunos, baseados no funcionamento do Sherlock:

- **Aluno 1 e aluno 2:** 50%
- **Aluno 1 e aluno 3:** 33,4%
- **Aluno 1 e aluno 4:** 25%
- **Aluno 1 e aluno 5:** 20%
- **Aluno 1 e aluno 6:** 16,7%
- **Aluno 1 e aluno 7:** 14,3%
- **Aluno 1 e aluno 8:** 12,5%
- **Aluno 1 e aluno 9:** 11,1%

Executou-se o Sherlock para a amostra criada, e o resultado é apresentado na Figura 1. Os seguintes valores foram utilizados nos parâmetros do software:

- **Zero bits (z):** 0, pois deseja-se que toda a assinatura seja considerada.
- **Chain length (n):** variado de 4 a 12, pois deseja-se que a assinatura seja criada utilizando de 4 até 12 palavras.
- **Threshold (t):** 0%, pois deseja-se que todo o resultado seja considerado.

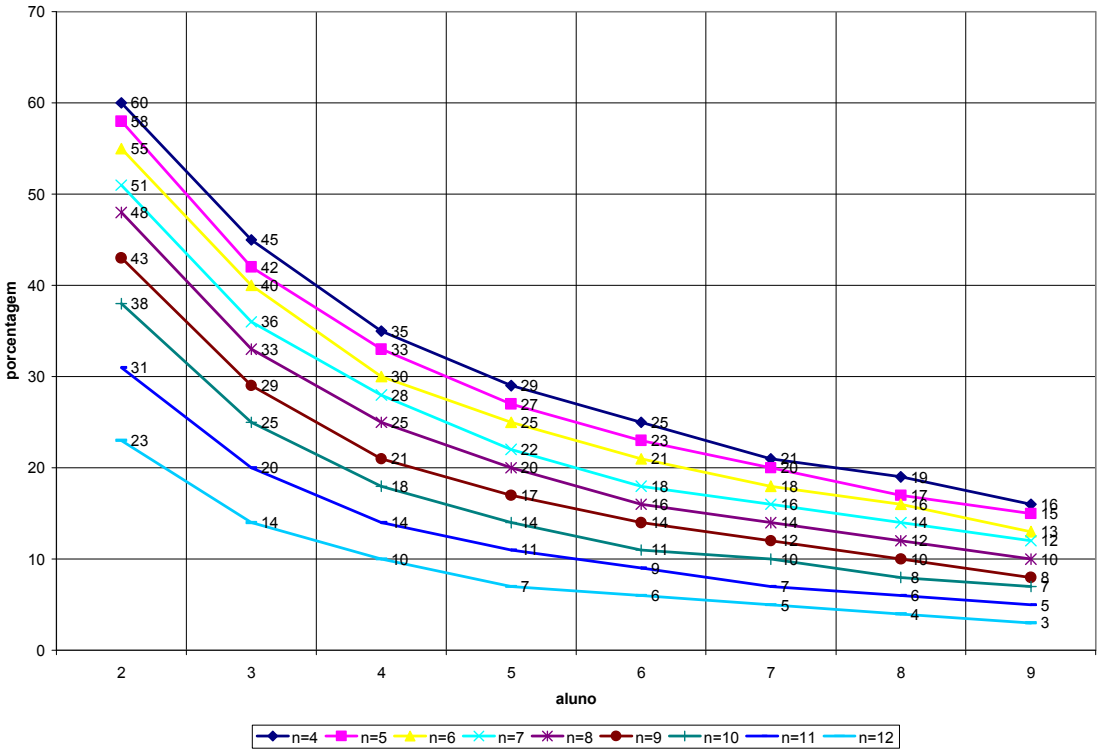

Figura 1 - resultado da avaliação

Nos resultados obtidos, nota-se que o valor encontrado não foi exatamente o esperado. Nota-se também que para os valores de n entre 4 e 6, os mesmos estão mais próximos do valor desejado do que os outros resultados onde n é maior do que 6.

5 Implantação

A implantação foi direcionada para uma aplicação voltada à internet utilizando a mesma tecnologia do TelEduc (NIED, 2008), o *PHP*. Uma interface para o usuário foi desenvolvida com o propósito de permitir tanto ao coordenador do curso quanto aos formadores do TelEduc selecionarem o curso e a questão desejada. O diagrama de atividades da interface é apresentado na Figura 2.

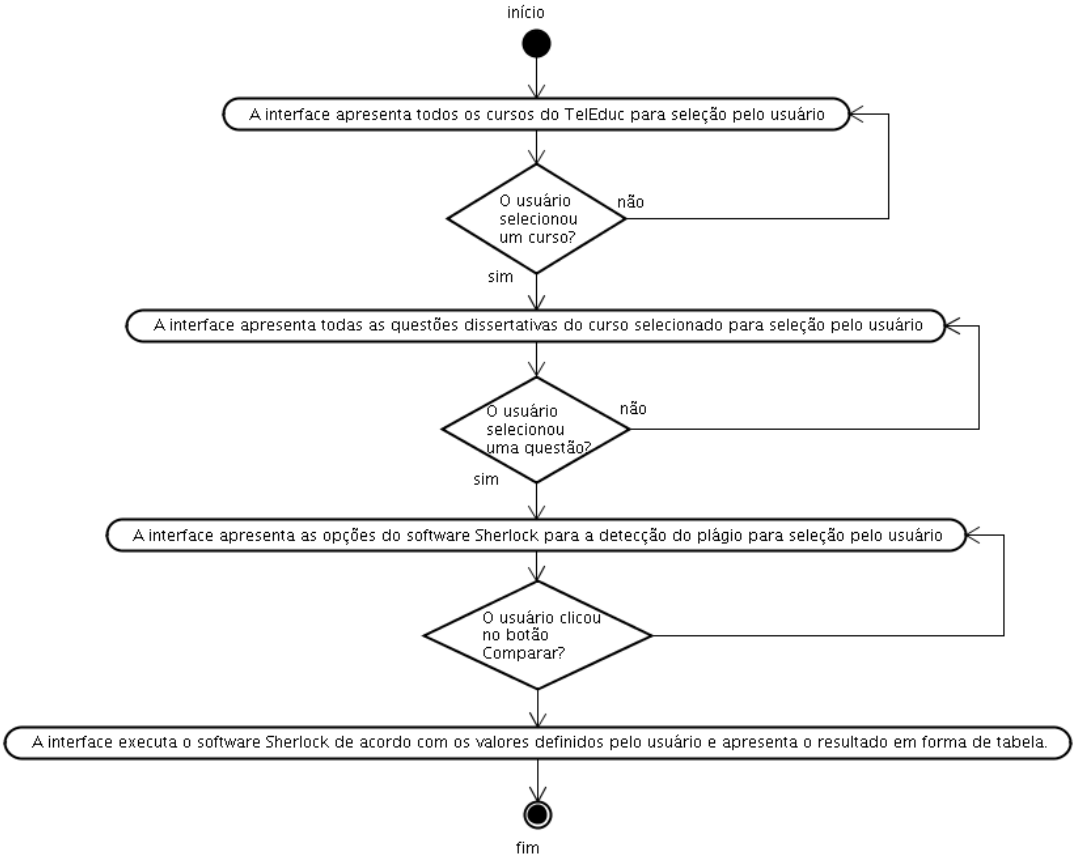

Figura 2 – diagrama de atividades da interface

No primeiro estágio, a interface pesquisa o banco de dados do TelEduc e apresenta para o usuário todos os cursos disponíveis para seleção.

Após a seleção do curso pelo usuário, a interface pesquisa quais são as questões dissertativas que foram aplicadas no curso escolhido e as apresenta para o usuário selecionar uma entre as encontradas.

Uma vez selecionada a questão, a interface apresenta as opções do software Sherlock e a interface está pronta para iniciar a comparação.

Por fim, ao usuário clicar no botão <comparar>, a interface executa o software Sherlock utilizando as opções desejadas pelo usuário e apresenta o resultado em forma de tabela. A Figura 3 apresenta a interface com a tabela comparativa.

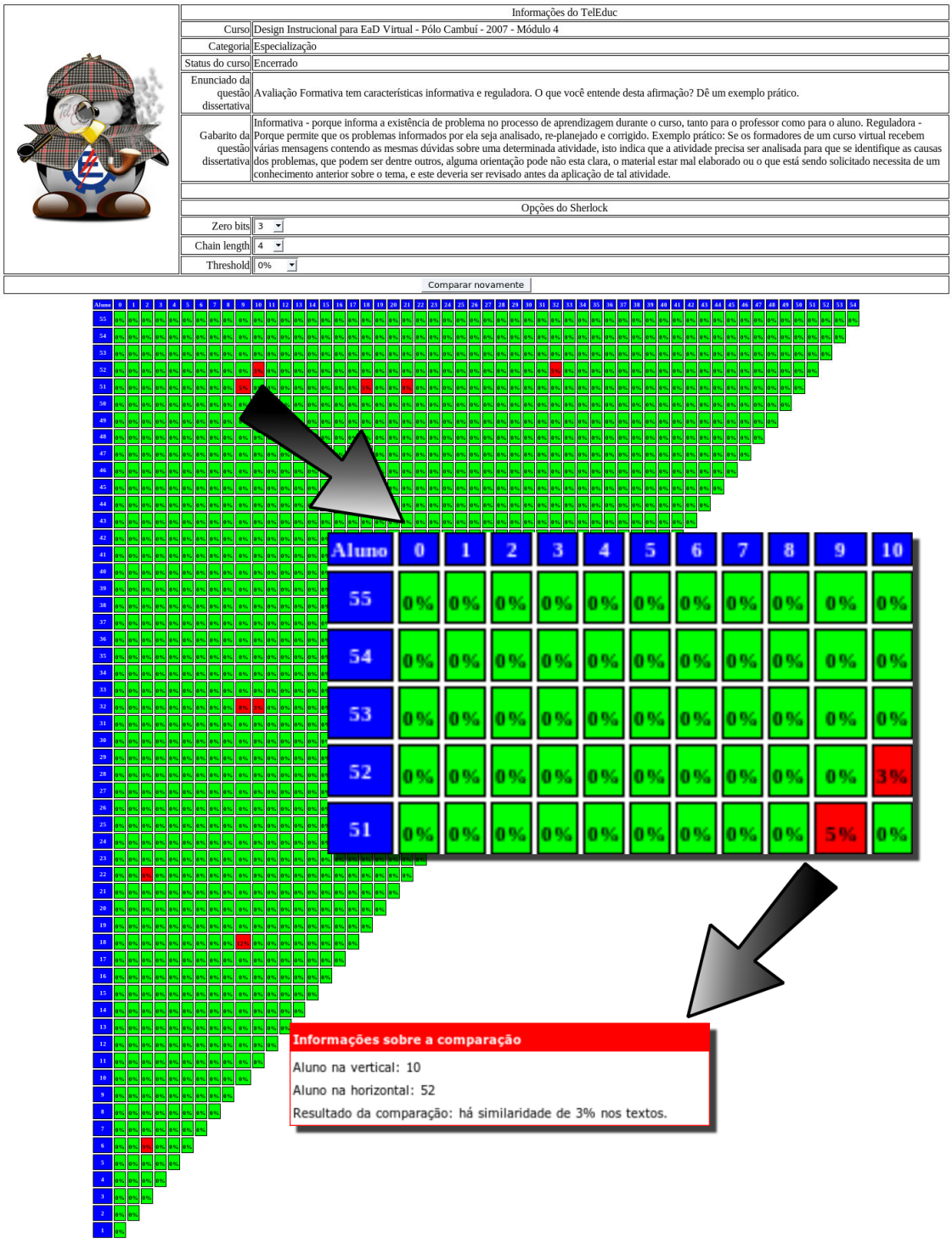

Figura 3 - a interface do sistema

Para facilitar a identificação na tabela comparativa, foi utilizada a cor verde para indicar a ausência de plágio e a cor vermelha, a porcentagem de plágio encontrada entre as duas respostas dos alunos. Quando o usuário deixar o cursor do mouse sobre o valor encontrado, uma mensagem informa quais são os alunos envolvidos e o índice encontrado. Este recurso é muito útil, pois o tamanho da tabela tende a ser muito grande, impossibilitando que seja mostrada em uma única tela.

6 Dados experimentais e resultados obtidos

Para verificar um caso real com o software Sherlock, foi utilizada uma questão dissertativa do curso de especialização em Design Instrucional para EaD Virtual - Pólo Cambuí - 2007 - Módulo 4, oferecido pela Educação a Distância da Universidade Federal de Itajubá. O curso contém 50 alunos e 6 tutores/formadores, totalizando 56 respostas à questão a ser analisada.

Partindo do número total de respostas e utilizando a Equação 5, obtém-se o número de iterações:

$$C\binom{56}{2} = \frac{56!}{2!\times(56!-2!)} = 1540$$

O software Sherlock teve seus parâmetros configurados da seguinte maneira:

- **Zero bits (z):** 3
- **Chain length (n):** 4
- **Threshold (t):** 0%

Após a análise, que durou cerca de 15 segundos para um microcomputador baseado no processador AMD Athlon X2 +6000 com 4 Gbytes de memória RAM e um disco rígido Serial ATA II de 7200 RPM, obteve-se os resultados apresentados na Figura 4.

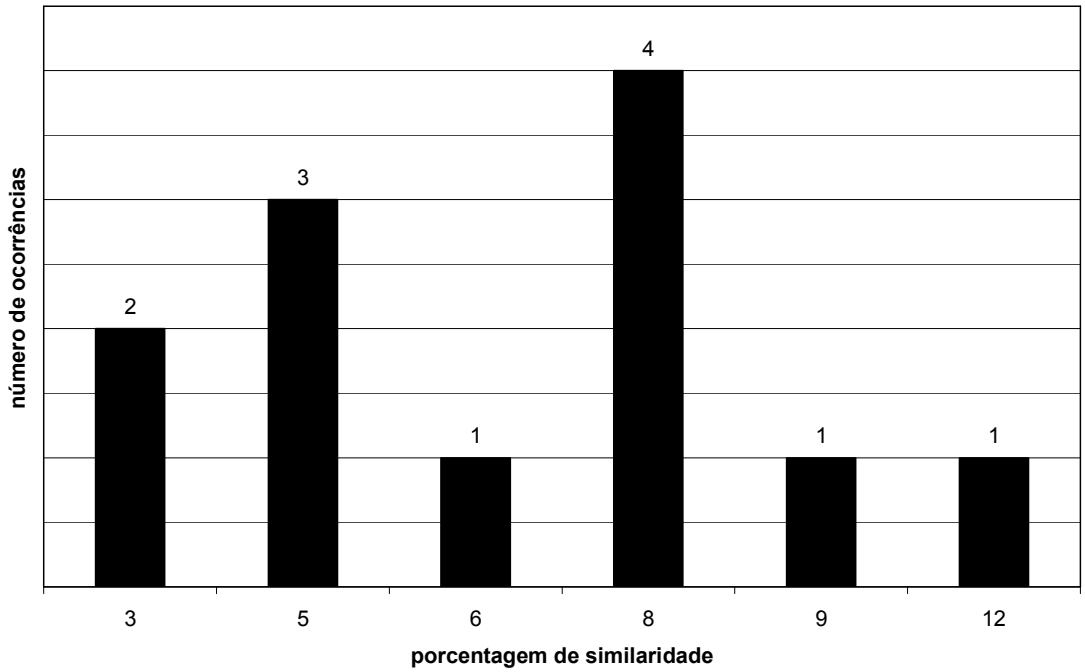

Figura 4 - resultado dos dados experimentais

7 Conclusão

O software Sherlock, foi testado através de uma amostra criada neste trabalho, apresentou uma resposta razoável no que diz respeito à detecção de plágio. De acordo com os resultados apresentados pela Figura 1, foi possível verificar a existência de grandes variações na detecção do plágio quando o parâmetro *chain length* é alterado. Antes de acusar o plágio, recomenda-se avaliar calmamente os dois textos envolvidos.

De acordo com os dados experimentais e os resultados obtidos do curso de especialização em Design Instrucional para EaD Virtual - Pólo Cambuí - 2007 - Módulo 4, verificou-se que o Sherlock encontrou uma baixíssima incidência de plágio, conforme apresentado pela Figura 4. Como o plágio buscado pela interface foi o intra-corpal, não foi possível avaliar a detecção do plágio extra-corpal na mesma amostra uma vez que a implementação dessa funcionalidade ainda não foi realizada. Porém, ao analisar o funcionamento do Sherlock, nota-se que o mesmo não possui outros tipos de verificação de plágio como, por exemplo, a substituição de palavras por sinônimos. Também não possui um tratamento especial dos textos que analise somente palavras que não possuem caracteres especiais latinos como, por exemplo, a cedilha ou o trema. Esta análise é particularmente muito importante, pois o Sherlock não será enganado pela omissão ou inclusão desses caracteres no texto, aumentando as chances de encontrar o plágio.

Como trabalho futuro, espera-se atingir estes objetivos através da implementação de todos esses recursos em *PHP* e o armazenamento das assinaturas em banco de dados, ao invés de utilizar os recursos já existentes do Sherlock para armazenamento das assinaturas. Como melhoria pode-se ainda otimizar o software para obter maior eficiência e velocidade do algoritmo através de técnicas utilizadas por White e Joy (2004), detectando plágio em linguagem natural. Outra melhoria seria o uso de técnicas de inteligência artificial propostas por Engels et al. (2007).

8 Referências bibliográficas

SCAIFE, B. Evaluation of Plagiarism Detection Software. **Plagiarism Detection Software Report For JISC Plagiarism Advisory Service,** Manchester, ver. 1.5, n. 11147, set. 2007.

ENGELS, S; LAKSHMANAN, V; CRAIG, M. Plagiarism detection using feature-based neural networks. **ACM SIGCSE Bulletin,** New York, v. 39, n. 1, p. 34-38, mar. 2007.

JOY, M; LUCK, M. Plagiarism in programming assignments. **IEEE Transactions on Education,** v. 42, n. 2, p. 129-133, maio 1999.

LIU, Y. et al. Extending web search for online plagiarism detection. **IEEE International Conference on Information Reuse and Integration,** Las Vegas, p. 164-169, ago. 2007.

MAURER, H; KAPPE, F; ZAKA; B. Plagiarism - A survey. **Journal of Universal Computer Science,** v. 12, n. 8, p. 1050-1084, ago. 2006.

NÚCLEO DE INFORMÁTICA APLICADA A EDUCAÇÃO - NIED, Universidade Estadual de Campinas - UNICAMP. **TelEduc**. Disponível em: <http://www.teleduc.org.br>. Acesso em: maio de 2008.

PIKE, R. **The Sherlock Plagiarism Detector**. Disponível em: <http://www.cs.su.oz.au/~scilect/sherlock>. Acesso em: maio de 2008.

ROSALES, F. et al. Detection of plagiarism in programming assignments. **IEEE Transactions on Education,** v. 51, n. 2, p. 174-183, maio 2008.

WHITE, D. R.; JOY, M. S. Sentence-based natural language plagiarism detection. **ACM Journal on Educational Resources in Computing**, Reino Unido, v. 4, n. 4, art. 2, dez. 2004.