

Propostas de Tese para os Mestrados em Informática (MI/MEI)

Pedro Rangel Henriques
Grupo de Processamento de Linguagens
DI/CCTC
Universidade do Minho

ano lectivo 09/10

1 Web Scrapping

Supervisor: Pedro Rangel Henriques + Daniela da Cruz

Área: extracção de informação da Web

Resumo:

A extracção de informação estática a partir de páginas Web é uma prática que tem vindo a crescer de dia para dia. Por exemplo, a colecção de "contactos" (moradas postais ou electrónicas, telefones, etc) existentes em vários sítios Web, a fim de os agregar numa única base de dados para posterior utilização com fim publicitário.

Neste projecto, pretende-se que automatize este processo de extracção. Para esse objectivo deverá criar uma representação abstracta da estrutura de uma página Web. Após isso, deverá criar uma ferramenta que permita indicar a colecção de sítios a considerar, a informação que se pretende extrair, como se quer agregar o resultado da extracção e definir também a forma de o exportar.

O trabalho de mestrado começará por uma extensiva pesquisa sobre ferramentas congéneres e sobre formas de representar uma página Web. No início deverão ser definidos alguns casos de estudo reais que permitam validar a aplicabilidade da solução final.

2 Reuse of Software Components

Supervisor: Pedro Rangel Henriques + Daniela da Cruz

Área: reutilização de código, análise de software, visualização

Resumo:

A reutilização de código tem sido aclamada nos últimos anos com a melhor política no desenvolvimento de novas aplicações. No entanto, na prática a reutilização não é uma tarefa fácil. A reutilização de um componente de software normalmente implica que certas restrições sejam observadas no sentido de preservar o correcto funcionamento do componente reutilizado e a sua integração segura no pacote em desenvolvimento. Essas restrições estão associadas a requisitos quer de forma, a nível do número e tipo de parâmetros ou do resultado, quer de conteúdo, a nível dos valores iniciais ou final desses parâmetros ou resultado. Tais restrições são actualmente tipicamente explicitadas, não só na linguagem de programação base, como em linguagens de anotação que permitem acrescentar ao código fonte pré ou pós-condições (por exemplo JML para Java, ou Spec# para C#).

De forma a melhorar as práticas de reutilização, esta proposta (que se insere no âmbito do projecto de pós-graduação "Gamma, um ambiente de análise e compreensão de programas") tem como objectivo criar uma ferramenta de ajuda à análise que permita identificar e mostrar todas as restrições a observar, sempre que se pretender reutilizar um componente. O sistema pretendido deve tornar-se mais inteligente, permitindo ver se o componente em apreço satisfaz uma determinada interface previamente especificada.

O trabalho de mestrado começará por uma extensiva pesquisa sobre ferramentas congéneres e sobre formas de apresentar ao utilizador a informação extraída. Especial cuidado deverá ser posto, no final, a mostrar o comportamento e usabilidade da ferramenta desenvolvida sobre um conjunto real de casos de teste de dimensão interessante; para isso, o aluno deve seleccionar no início a linguagem base com que pretende trabalhar (C#, Java, ou outra).

3 Automatic extraction of Ontologies from the Problem Domain

Supervisor: Pedro Rangel Henriques + Maria João Varanda + Daniela da Cruz

Área: compreensão de programas, ontologias, extracção/representação de conhecimento

Resumo:

A extracção automática do modelo das aplicações a partir da análise do código-fonte tem sido prática corrente ao longo dos últimos anos. No entanto se tal se faz sem grande dificuldade ao nível do domínio do programa, o mesmo não acontece em relação ao domínio do problema.

Tendo o domínio do problema um papel crucial na compreensão de programas, pretende-se com esta proposta de tese (que se integra no projecto de investigação "NAU: Exploração formal do Domínio do Problema e criação de um mapeamento para navegação entre os Modelos do Domínio do Problema e do Programa com vista a melhorar a Compreensão de Programas") extrair automaticamente informação relevante para a compreensão de uma aplicação, a partir de descrições do domínio do respectivo problema. Esta informação extraída deverá ser representada na forma de uma ontologia, visando a melhor percepção dos conceitos envolvidos no problema.

Isto requer que no início se caracterize, o mais amplamente possível, o termo *domínio do problema* e se identifiquem as fontes a partir das quais se pode construir a referida ontologia. Devem também ser estudadas notações para representação e visualização de ontologias, de modo a fundamentar a opção que o aluno tomar. O sistema de exploração e extracção de ontologias deve ser testado em casos reais, dentro de uma bateria de testes a escolher no início do projecto.

4 Analysis and Evaluation of Open Source Software Quality

Supervisor: Pedro Rangel Henriques + Daniela da Cruz

Área: análise de código, avaliação de software

Resumo:

A avaliação da qualidade de software é crucial quando se fala em usar software aberto, ou reutilizar algumas partes, para o desenvolvimento de novas aplicações. No entanto, não existem ainda normas que facilitem esta avaliação.

Esta proposta de tese, que surge no âmbito do projecto de investigação financiado pela FCT "CROSS, uma infraestrutura para a certificação e re-engenharia de software livre", visa criar um conjunto coeso de métricas que permitam tirar algumas conclusões acerca da qualidade de um pacote em apreço. Como base de trabalho e para testes, serão usados pacotes retirados de fontes bem conhecidas: SourceForge, CodePlex, Google Code, etc.

Este trabalho começará por uma extensa pesquisa sobre métricas de software. A sua utilidade final deve ser comprovada com a análise de pacotes reais extraídos das fontes citadas.

5 Clone Detection based on Patterns

Supervisor: Pedro Rangel Henriques + Daniela da Cruz

Área: análise de código, clonagem, padrões de software

Resumo:

Em projectos de grande dimensão, várias vezes as mesmas classes/funções são usadas para diferentes funcionalidades. Outras vezes, os vários intervenientes vão criando, ao longo do desenvolvimento de um projecto, diferentes funções com a mesma funcionalidade.

Esta proposta de tese, que surge no âmbito do projecto de investigação financiado pela FCT "CROSS, uma infraestrutura para a certificação e re-engenharia de software livre", visa detectar clones de código com base em padrões.

Estes padrões (que serão dados pelo utilizador ou gerados pela própria ferramenta de análise) poderão ser por semelhança de estrutura ou baseados na análise de nomes (mais conhecido por *naming analysis*).

O trabalho de mestrado começará por uma extensiva pesquisa sobre ferramentas congéneres e sobre padrões (caracterização precisa, formas de os representar, catálogos já existentes, etc.). No início deverão ser definidos alguns casos de estudo reais que permitam validar a aplicabilidade da solução final.

6 Annotation Inference based on Function Calls

Supervisor: Pedro Rangel Henriques + Daniela da Cruz

Área: análise de código

Resumo:

A programação baseada em contratos (*Design-by-Contract*) desempenha um papel importante na garantia da correcção de um programa. No entanto, não tem sido um conceito vulgarmente colocado em prática devido à dificuldade, não só na aprendizagem das linguagens de anotação (JML, Spec#, Caduceus, ASML, etc), como também na escrita das próprias condições (pré-condições, pós-condições, invariantes).

No âmbito do projecto de pós-graduação "Gamma, um ambiente de análise e compreensão de programas", propõe-se este tema de tese que tem como objectivo facilitar a utilização de anotações num programa. Esta proposta visa automatizar a inferência de anotações (provavelmente assistida pelo programador) com base no estudo interno da função e também com base num conjunto dado pelo utilizador de chamadas à função que se pretende anotar.

A parte de implementação será precedida de um cuidadoso estudo sobre anotação de programas com pré e pós-condições e sobre a existência de ferramentas similares. A validação da aplicabilidade da solução final a sobre casos de estudo reais é uma exigência da proposta.

7 Analysis and Visualization of Dependencies in Intermediate Level Code

Supervisor: Pedro Rangel Henriques + Daniela da Cruz

Área: análise de código, visualização/compreensão de programas, slicing, grafo de dependências

Resumo:

O estudo das dependências entre variáveis ou do fluxo de execução das instruções a nível do código fonte é uma técnica de análise muito em voga, há muitos anos, e muito usada na compilação, optimização e compreensão de programas. Mas com o advento de plataformas de trabalho integradoras de linguagens diferente (como é o caso do .Net e outras) passam a ter um especial relevo as linguagens intermédias (como, p.ex., o CLI); nesse contexto, muitos módulos integrados em aplicações reais só estão disponíveis nessa representação intermédia. Assim o estudo e análise de código intermédio de uma aplicação é uma necessidade real e urgente.

No âmbito do projecto de pós-graduação "Gamma, um ambiente de análise e compreensão de programas", propõe-se este tema de tese que tem como objectivo explorar as dependências entre instruções ao nível intermédio, de modo a construir grafos de fluxo de controlo e de dependência de dados para CLI, sendo capaz de relacionar as instruções intermédias com o respectivo código fonte. O ambiente de análise a desenvolver deverá ser capaz de fazer o slicing do código intermédio, usando os grafos de dependências construídos na fase anterior.

Como trabalho de arranque, este projecto requer o estudo de uma linguagem intermédia (de preferência o CLI) e de técnicas de slicing. A validação da aplicabilidade da solução final a sobre casos de estudo reais é uma exigência da proposta.

8 Reasoning about the quality of UML constrained models

Supervisor: Pedro Rangel Henriques + Daniela da Cruz
Área: análise de código, modelação de SW, qualidade

Resumo:

Nesta proposta de tese, que surge no âmbito do projecto de pós-graduação "Gamma, um ambiente de análise e compreensão de programas", pretende-se fazer um estudo de métricas e técnicas de visualização para análise da qualidade de modelos UML estendidos com restrições escritas na linguagem OCL, identificando-se a existência de ferramentas similares. Com base nesse estudo deve ser feita uma proposta, passando-se então à fase de implementação de uma ferramenta de ajuda à compreensão de modelos que deverá ser devidamente validada sobre casos de estudo reais.

9 Semantic-based Tools for Problem Solving Assessment (Proposals I/II)

Supervisor: Pedro Rangel Henriques, Daniela da Cruz
Área: análise de código, modelação de SW, qualidade

Resumo:

In the context of the FCT founded MathIS, or Matisse, project, the main contribution of these two proposals is a set of tools to assist teachers, or generally speaking jurists, in semantic code assessment of algorithms for problem solving created, coded and automatically submitted by students (or, generally, candidates) that, in some learning or contest environment, are asked to solve problems. The assessment system will take as input the solution produced by the program developed by the candidate (in response to the proposed exercise) and some description of the problem to be solved or the expected solution, which will come up with a final statement about the correctness of the given solution. The different approaches that can be followed to verify the compliance of the produced solution with the expected one, give rise to different sub-tasks and to a set of support tools. More specifically we want to explore three ways (two in the first thesis, and one in the second) to do the assessment. Our aim for this Task is to automate this verification process, but adopting a semantic approach that will allow us to process a broader class of problems. For this we shall resort to language processing techniques, based on a grammatical approach, to explore different ways to validate the answer produced by the submitted programs. The envisaged approaches are:

- Approach 1 consists in the use of an internal abstract representation; expected and produced answers are compiled into that representation and the comparison is done at that level.
- Approach 2, instead of taking as input a formal description of the expected answer, takes a mathematical description of the algorithm and generates the output to compare with the produced answer.
- Approach 3 is based on the interpretation of the answer by a virtual machine that executes it as a program. In this case also animation techniques can be used to aid in analyzing and assessing the answer.

The exploration of Approaches 1 and 2 will constitute one thesis, while Approach 3 is the subject of a second thesis.