Atas das Sétimas Jornadas de Informática da Universidade de Évora

Évora, 7 de Abril de 2017

UNIVERSIDADE DE ÉVORA

Paper/Author Index

Transformação Digital na Área Hospitalar

(Revisão do Estado da Arte)

John Acevedo Nanclares

(jjananclares@gmail.com)

Resumo: A transformação digital na saúde tem acontecido de forma irregular. Nos equipamentos a evolução digital tem sido gigante e de total aceitação. Nos sistemas de apoio à decisão clínica, o desenvolvimento de aplicações tem sido significativo, mas de difícil aceitação. O modelo de saúde tem-se mantido sem grandes alterações apesar das suas premissas terem mudado. A especialização clinica e a medicina do retalho obrigarão a repensar o modelo, evoluindo para uma organização de trabalho em rede e avaliado pelos resultados obtidos na solução dos problemas de saúde e não pela realização de atos clínicos isolados. A grande transformação digital nesta área passa pela construção de um modelo de "colaboração" com o objetivo de fazer a gestão da saúde: preventiva, preditiva, curativa e de convivência com a doença, suportado em ferramentas informáticas que permitam fazer diagnósticos mais precisos e rápidos e terapêuticas personalizadas mais eficazes, incrementando a eficiência do ecossistema e facilitando-lhe a vida aos doentes.

Palavras chave: Transformação Digital Saúde, Informatização Hospitalar, Ferramentas de Suporte à Decisão Clinica, Health Machine Learning, Health Big Data, Health Apps, Medicina Personalizada, Medicina Preventiva, Monitorização Remota de Doentes, Gestão da Saúde

1 Introdução

A virtualização de operações nas organizações tem sido uma constante de desenvolvimento à qual temos assistido aceleradamente nos últimos anos. Ferramentas informáticas móveis inundam o mercado em todos os setores da nossa sociedade facilitando as tarefas cotidianas dos cidadãos. Neste trabalho far-se-á uma revisão do grau de penetração que as ferramentas informáticas têm tido no setor da saúde com o foco de análise nas unidades hospitalares. Para cada processo relevante dentro do âmbito hospitalar será feita uma descrição das ferramentas informáticas que o suportam, assim como uma descrição da linha de orientação que conduz o seu desenvolvimento. Grandes investimentos são feitos na informatização do setor da saúde [1], mas é preciso fazer uma análise crítica do valor acrescentado que estas iniciativas trazem. Uma reflexão sobre os aspetos de maior aceitação e os principais obstáculos a esta transformação digital ajudaria a definir qual a estratégia mais apropriada a seguir no setor da saúde, sobretudo no contexto hospitalar (quer para a sua implementação, quer para o seu aproveitamento).

2 Metodologia

A realização deste trabalho baseou-se numa revisão bibliográfica de artigos científicos disponíveis nos motores de busca do Google Académico e B-on. Além da consulta bibliográfica de caráter científico foram tidos em conta documentos que caraterizam os planos estratégicos de sistemas de informação de diferentes organizações hospitalares.

3 Objetivos

O paradigma da Transformação Digital tem sido o fator essencial nas grandes mudanças acontecidas na sociedade contemporânea. A análise do impacto deste fenómeno na área da saúde e particularmente no âmbito hospitalar é o objetivo principal deste trabalho. Olhar para os principais processos hospitalares e identificar quais as ferramentas informáticas que são usadas e o seu grau de aceitação dentro dos diferentes utilizadores é o propósito desta reflexão.

4 Transformação Digital na Área da Hospitalar

O contexto hospitalar ou de unidades de prestação de serviços de saúde, independentemente de qual seja a sua dimensão ou tipologia (pública ou privada), é suficientemente amplo para ser recomendável uma divisão nas suas componentes mais características: clínica e gestão.

Neste trabalho relacionado com o estado da arte da informatização na área hospitalar vamos estabelecer esta divisão e vamos propor para cada uma das vertentes, clínica (Clínica-Clínica) e Gestão (Gestão-Clínica), um macro-circuito que identifica os processos mais importantes dentro de cada uma destas áreas.

Partindo da identificação de necessidades de saúde por parte dos cidadãos vamos estabelecer o percurso que é seguido por estes ao longo do circuito de resolução dos seus problemas. Vamos diferenciar os processos do âmbito Clínico-Clínico onde os protagonistas da realização das atividades de cada processo nesta vertente, para além dos doentes, são os profissionais de saúde. E os processos do âmbito Gestão-Clínica onde os protagonistas da realização das atividades inerentes a cada um dos processos de esta outra componente são, para além dos doentes, os gestores e pessoal administrativo das unidades de saúde.

No circuito Clínico-Clínico destacamos os seguintes processos:

- Identificação da Necessidade: a pessoa apercebe-se de um problema de saúde para o qual não tem a capacidade de resolver por si própria e precisa da ajuda de profissionais de saúde para ser tratada.
- Elaboração do Diagnóstico e Definição da Terapêutica: os profissionais de saúde com base no seu conhecimento e experiência, tendo em conta os protocolos de diagnóstico e terapêutica instituídos, elaboram o diagnóstico e definem o plano de tratamento a seguir para o doente.
- Realização do Tratamento: é realizado pelo doente com ou sem ajuda de profissionais de saúde através da execução do plano de tratamento: prescrição medicamentosa, atos de enfermagem.
- Controlo da Evolução do Doente: é realizado pelos profissionais de saúde no caso do doente se encontrar nas instalações hospitalares ou pelo próprio doente no caso de se encontrar fora das instalações de saúde com ou sem intervenção de profissionais de saúde.
- Plano de Prevenção e Predição: é realizado pelas próprias pessoas recorrendo de forma pontual aos profissionais de saúde.

No circuito Gestão-Clínica destacamos os seguintes processos:

- Identificação da Necessidade: a pessoa apercebe-se de um problema de saúde para o qual não tem a capacidade de resolver por si própria e precisa da ajuda de profissionais de saúde para ser tratada.

- Pedidos de Informação/Procura/Marcação: é feita uma avaliação por parte do doente de quais são as alternativas que existem para a solução do seu problema para seguidamente fazer o respetivo agendamento.
- Check-In/Check-Out: é feito pelo próprio doente com ajuda do pessoal administrativo da unidade hospitalar e corresponde ao percurso que o doente faz desde a sua chegada à unidade de saúde até à sua saída.
- Ato clínico: é realizado pelos profissionais de saúde e referimo-nos aqui não à componente clínica, mas à componente administrativa e logística da realização do ato clínico.

Seguidamente, faremos a caraterização informática de cada um dos processos identificados para cada um dos circuitos: Clínico-Clínico e Gestão-Clínica. Apresentaremos um ponto de situação do atual grau de uso de ferramentas digitais em cada processo e a seguir definiremos uma linha de ação futura que se prevê será o caminho que devera conduzir a evolução digital em cada um dos mesmos.

4.1 Circuito Clínico-Clínico

4.1.1 Elaboração do Diagnostico e Definição da Terapêutica

Descrição: A área do diagnóstico clínico é uma das áreas mais evoluídas no uso de ferramentas digitais para a sua realização. Os elementos principais para a sua execução são a componente analítica, sustentada na semiologia clínica e a área de meios complementares de diagnóstico.

Na componente analítica o uso de ferramentas como Bases de Dados Clínicos, Bases de Dados de Medicamentos, Histórias Clínicas Digitalizadas (EPR) é de caráter cotidiano e de aceitação comum por parte dos profissionais de saúde. É normal a utilização destas ferramentas informáticas nos meios hospitalares como clínicas, hospitais, centros de saúde e até nos consultórios privados de pequena dimensão. Neste sentido as ferramentas informáticas nesta área são já consideradas commodities e tem o seu futuro assegurado como pilares de evolução do diagnóstico clínico.

Já não podemos afirmar o mesmo no que se refere a ferramentas informáticas orientadas ao apoio da decisão clínica tais como Sistemas Periciais Clínicos (inteligência artificial), machine learning e big data. Apesar de estar em clara ascensão no que se refere à investigação e desenvolvimento, o seu uso ainda se encontra aquém das expectativas tendo em conta a sua grande potencialidade e os esforços financeiros realizados neste campo. É evidente que na decisão clínica existe uma componente subjetiva que é reservada agora e o será no futuro aos profissionais de saúde, mas como em todas as disciplinas científicas existe também uma componente algorítmica suscetível de ser abordada com modelos matemáticos e probabilísticos que a seguir podem ser informatizados garantindo assim a sua estandardização e reprodutibilidade. Esta componente ganha cada dia mais terreno estreitando a faixa da subjetividade na decisão clinica. O uso das ferramentas de machine learning associadas a grandes volumes de informação (big data) permitem aos investigadores e profissionais de saúde ter à mão informação em volume, qualidade e rapidez de tal forma que as suas decisões podem ser diminuídas em probabilidade de erro, assim como ser orientadas aos doentes de forma personalizada. Conceitos como Medicina Baseada na Evidencia ou Medicina Personalizada (precision health) ganham força com o aparecimento e desenvolvimento destas ferramentas [2] [3].

Temos ainda uma cultura, que apesar de não ser idêntica em todo o mundo, tem em comum a sua

falta de aceitação por parte dos profissionais de saúde deste tipo de ferramentas [4], o que tem contribuído para a modesta banalização do seu uso, não permitindo ainda ser possível a criação do circuito de interações de desenvolvimento (desenho-prototipagem-desenvovimento-testes-produção-afinação-detecção de erros-desenho...) fundamental para evolução rápida de qualquer modelo. A utilização tímida destas ferramentas tem travado o seu ritmo de evolução, mas é evidente que é esse o caminho a seguir e mesmo de forma lenta tem avançado a passo firme nos últimos anos e continuará sem parar.

Na componente de meios complementares de diagnóstico, principalmente no que diz respeito a imagiologia, a transformação digital tem sido notável nos últimos anos [2]. Equipamento de radiologia e ferramentas como PACS com as sua poderosas consolas de diagnóstico têm contribuído para uma evolução sem precedentes. Aqui são os próprios médicos radiologistas que servem de promotores e incentivadores às empresas de investigação, construção de equipamentos e desenvolvedores de software para estes continuarem de forma exponencial a disponibilizar soluções. A aceitação destas ferramentas acontece neste meio de forma natural e poderíamos falar com maior rigor de evolução digital, mais que de uma transformação digital.

Na patologia clínica também é evidente o uso cotidiano de ferramentas digitais, mesmo que não ao nível de imagiologia, esta área também tem sofrido grandes mudanças derivadas da evolução tecnológica, principalmente nos equipamentos de medição.

Perspetiva Futura: Com a Genómica como linha de orientação predominante na investigação médica [5], o desenvolvimento de uma medicina personalizada torna-se uma evidência. Ter a possibilidade à partida ou com o tempo suficientemente confortável para agir, de detetar potenciais focos de problemas em cada individuo em particular, tendo em conta a sua estrutura genómica, vai ser a grande revolução na área clínica. A medicina preditiva e preventiva serão os focos de investimento a todo nível dentro do contexto da saúde. Aumentamos aqui o grau de precisão com que os diagnósticos podem ser feitos, diminuindo a probabilidade de erro e estreitando a faixa de subjetividade associada a esta tarefa, o que leva ao aumento da componente determinística na realização de diagnósticos e à consequente possibilidade de criar/afinar algoritmos de auxilio a esta atividade. Tudo o referido anteriormente promoverá uma mudança de atitude por parte dos profissionais de saúde na aceitação de ferramentas de ajuda na decisão clínica baseada em sistemas de inteligência artificial, utilizando big data e ferramentas de machine learning numa medicina virada para a prevenção e predição e baseada na evidencia clínica [6].

4.1.2 Realização do Tratamento

Descrição: O tratamento de doentes não é uma área exemplar da transformação digital ocorrida na área da saúde. Podemos, no entanto, destacar o tratamento feito dentro das unidades hospitalares (tanto no internamento como no ambulatório) onde a digitalização chega através dos equipamentos cada vez mais automatizados e cujo uso é de aceitação natural e até diferenciadores em termos de concorrência entre unidades de saúde.

No tratamento feito pelos doentes fora das unidades de saúde, em casa, a ausência de ferramentas digitais facilitadoras da execução deste trabalho é quase total. Salientamos só o aparecimento nos

últimos anos de apps que permitem aos utilizadores a gestão de toma de medicamentos e a realização de atos de enfermagem e fisioterapia.

Consideramos, portanto, o processo do tratamento uma área fértil para a investigação e desenvolvimento de ferramentas digitais que facilitem aos doentes, fora das unidades hospitalares, a possibilidade de fazer todo o processo de recuperação de forma mais ágil, segura e até controlada por profissionais de saúde, mesmo que de forma remota.

Perspetiva Futura: À medida que a excelência clínica se torna um fator diferenciador entre unidades de saúde e entre organizações prestadores de saúde de uma forma mais geral (pública e privada) e até um fator exigível tanto a nível de políticas de saúde e principalmente pelos utilizadores que cada vez estão mais apurados na literacia da saúde, surge como uma obrigatoriedade a medição mais rigorosa e ampla dos Outcomes Clínicos ao longo da Cadeia de Valor da Saúde. Medições que não se esgotam após a realização de atos clínicos (como por exemplo cirurgias) e/ou durante o período de estadia dos doentes nas unidades hospitalares, mas sim, durante todo o período de recuperação do mesmo, incluindo o período transcorrido fora das unidades (em casa) até à sua recuperação "total". O aparecimento de ferramentas de gestão da recuperação de doentes fora das unidades hospitalares é o caminho traçado para a transformação digital na vertente do Tratamento. Dispositivos de hardware e software, como Portais de Interação Doente-Unidade on-line, ou quase on-line onde se faça o controlo da evolução do doente vs. modelo de terapêutica e medicamentos utilizados, parece ser o passo seguinte neste percurso.

4.1.3 Controlo da Evolução do Doente

Descrição: Para falarmos de Controlo devemos primeiro falar de Monitorização. Referimo-nos aqui à Monitorização Remota, ou seja, ferramentas que permitam a medição de variáveis à distância. Podemos separar 3 tipos de monitorização: monitorização de doentes agudos, monitorização de doentes crónicos e monitorização com fins especiais tais como desportistas, grávidas, viajantes, expatriados e outros.

A monitorização de agudos refere-se a doentes que devendo permanecer internados nas unidades de saúde, são enviados para casa para completar o seu período de internamento sob a responsabilidade da unidade de saúde respetiva. Este é um modelo ainda muito embrionário e carente de mecanismos digitais de monitorização. Mesmo assim, estão dando-se os primeiros passos neste sentido através do desenvolvimento de unidades de cuidados intensivos portáteis e farmácias (unidose) portáteis interligadas a sistemas de controlo hospitalar que permitam ter o doente suficientemente vigiado para garantir a sua segurança e permitir uma oportuna intervenção dos profissionais de saúde no caso ser necessário e assegurar também a devida administração da terapêutica medicamentosa (esta monitorização poderia encaixar também na etapa do tratamento).

A monitorização de doentes crónicos é uma das áreas de maior foco nestes momentos. O aparecimento de equipamentos portáteis com fins específicos, como medidores de variáveis diversas (tensão arterial, ECG, glicemia, colesterol, etc.) é uma realidade [2]. Estes dispositivos já estão dotados de mecanismos de armazenamento de informação e transmissão de dados que permitem aos doentes e entidades prestadoras de serviços de saúde manter uma permanente comunicação,

monitorização e consequentemente controlo das mais diversas situações relacionadas com saúde dos doentes crónicos. O aparecimento de apps ligadas a estes equipamentos pertence também ao cotidiano, estas apps, inclusive, podem trazer embebidos algoritmos de terapêutica de tal forma que já há registos de profissionais de saúde a prescrever apps em vez de medicamentos com a sua devida customização ou adaptação à pessoa em particular. Mesmo assim há muito por evoluir neste contexto e o nível de aceitação deste tipo de ferramentas ainda não é o suficiente para permitir uma completa banalização do seu uso e consequentemente a velocidade de disseminação destas é ainda lento. Fazem parte deste tipo de equipamentos todos os dispositivos que estão a aparecer no mercado com a caraterística principal de serem totalmente digitais e suficientemente ergonómicos para facilitar o seu uso cotidiano (wearables).

A monitorização com fins especiais já se faz algum tempo mesmo que as ferramentas usadas para este fim estejam cada vez mais evoluídas, tal como vimos na monitorização de doentes crónicos. A colocação de sensores em varias partes do corpo com o objetivo de predizer situações anómalas e afinar planos de treino é já comum no contexto do desporto de alto rendimento, por exemplo.

Perspetiva Futura: Os avanços em nanotecnologia, que têm sido gigantes nos últimos anos, irão permitir a conversão dos equipamentos de medição/ação de variáveis clínicas em dispositivos diminutos com as mesmas funcionalidades e facilmente implantáveis no corpo dos doentes, de tal forma que o nível de automatismo será quase total proporcionando aos doentes uma monitorização em tempo real e sem a necessidade de intervenção manual. Nesta área podemos também falar do mesmo tipo de avanços, mas aplicados ao setor do medicamento no qual vai-se poder também ter à disposição nano-dispositivos de administração de medicamentos implantáveis no corpo dos doentes, eliminando a necessidade da intervenção humana.

4.1.4 Plano de Prevenção e Predição

Descrição: Derivado do sucesso contínuo na investigação médica e farmacêutica, a expectativa de vida a nível geral tem aumentado de forma acentuada nas ultimas décadas. As consequências desta mudança são de grande impacto num ecossistema sustentável de saúde a nível mundial, transferindo o peso das doenças agudas para as doenças crónicas. Isto significa que uma pessoa cuja expectativa de vida à nascença é de X anos, vai passar muitos mais anos a conviver com uma ou mais doenças crónicas. Para além de todas as desvantagens a nível pessoal e social que esta situação tem, a capacidade financeira das pessoas, famílias e do estado torna-se insustentável; daí que os sistemas de saúde a nível mundial se encontrem quase todos deficitários recorrendo a grandes ginásticas para garantir a sua difícil sobrevivência.

Isto evidencia a necessidade de uma mudança de paradigma na maneira como é percecionada e gerida a saúde a nível mundial. A aposta na prevenção e a predição surgem como uma alternativa ao modelo atual [6]. Mesmo assim, e com todas as evidências à mostra, o enveredar por este caminho tem sido feito com exagerada timidez. Só agora com o desenvolvimento da intercomunicabilidade social (smartphones, redes sociais, wearables) estão dadas as condições para acelerar este percurso. Volta a ser a tecnologia o grande promotor de uma mudança social fornecendo as condições necessárias à criação de ferramentas informáticas que permitam apostar na prevenção em saúde.

É cada vez maior aparecimento de sites orientados a instruir a população para hábitos mais saudáveis, bem como o aparecimento de apps para registar e controlar a atividades desportivas e ainda para sugerir o seguimento de dietas alimentícias e promover uma série de atividades profiláticas.

Perspetiva Futura: Semelhante à evolução experimentada em outros domínios como por exemplo no desporto, no qual existem os treinadores pessoais ou na Banca onde a relação interpessoal entre os clientes e a instituição se faz através dos gestores de clientes, na saúde o caminho vislumbra-se parecido. Prevê-se o surgimento de Gestores Pessoais de Saúde, que terão como objetivo o aconselhamento e controlo da saúde dos cidadãos. Obviamente tudo suportado em ferramentas informáticas (algumas já existentes) que facilitarão este trabalho tanto na vertente de prevenção como de convivência com a doença.

O Livro da Saúde de cada individuo será o seu guião a seguir, formatado inicialmente como um livro de recomendações de caráter determinístico baseado em variáveis explícitas associadas a cada indivíduo, mas posteriormente evoluirá para um livro de recomendações baseado na análise de grandes volumes de informação associada ao indivíduo confrontada com grandes volumes de informação existentes em bases de dados a nível mundial operado por um motor de inteligência artificial.

4.2 Circuito Gestão-Clinica

4.2.1 Pedidos de Informação/Procura/Marcação

Descrição: Como todas organizações atuais, as unidades de saúde fazem o seu primeiro contato digital com os seus utilizadores através do Site da instituição. É por este meio que se subministram todas as informações relevantes tanto a nível institucional (Missão, Visão, Relatórios de Contas e outros), como a nível operacional (Serviços, Produtos, Agendas, Preços, Acordos e outros). Em outro estádio estão aquelas unidades de saúde que tem evoluído na interação com os seus clientes proporcionando-lhes informação clínica, nomeadamente: resultados de exames de patologia clínica, relatórios de imagiologia e relatórios de visitas à unidade e permitindo ações administrativas tais como marcações de atos clínicos e pagamentos. São estes os níveis de digitalização em que as unidades de saúde se encontram no que diz respeito a este processo. O que revela uma discreta utilização das tecnologias de informação na interação entre unidades de saúde e clientes.

Verificamos que os modelos de gestão (já implementados em outros domínios) que permitissem a possibilidade de fazer comparações entre unidades de saúde e a possibilidade de fazer avaliações mútuas entre estas unidades de saúde e os seus clientes ainda não se encontram banalizados. Assim como não existe a possibilidade de fazer comparações entre profissionais de saúde que pertencem a uma mesma especialidade ou que realizam um mesmo ato clínico; como também não existe a possibilidade de fazer avaliações mútuas entre estes profissionais de saúde e os seus clientes.

No momento da procura de alternativas, depois de detetada a necessidade de recorrer um profissional de saúde ou uma entidade de saúde é evidente a falta de ferramentas informáticas que suportem esta tarefa. A procura é feita de uma forma tradicional, não estruturada e portanto, não informatizada.

Perspetiva Futura: Se tomarmos como exemplo a consulta, marcação e compra de bilhetes aéreos, vemos quão atrasada está a área da saúde neste domínio. Vamos assistir rapidamente à disponibilização de ferramentas informáticas com motores de busca que permitirão aos utilizadores procurar o melhor fornecedor para as suas necessidades pontuais de saúde e com os filtros que sejam relevantes para cada pessoa como geografia, excelência clínica, preço e disponibilidade, entre outros.

Ferramentas administrativas informatizadas que permitam a protocolização das medições de outcomes Clínicos, ligadas a ferramentas já existentes como CRM também estão na linha de orientação do desenvolvimento digital das organizações de saúde.

4.2.2 Check-In/Check-Out

Descrição: Existem dois circuitos básicos no contexto das unidades de saúde: o circuito do doente e o circuito do medicamento. O circuito do medicamento faz parte da componente logística das organizações de saúde e encontra-se razoavelmente informatizado dentro do Sistema de Informação do Hospital (HIS).

Focar-nos-emos aqui no circuito do doente, desde a sua chegada às unidades de saúde até a sua saída. Quando um doente chega a uma unidade de saúde o primeiro que faz é dar conhecimento da sua chegada através de uma ação chamada check-in que pode ser realizada de forma manual informando o administrativo responsável da sua chegada ou fazendo check-in automático em dispositivos próprios para tal efeito. Nesta tarefa são fornecidas todas as informações de caráter administrativo (Identificação, Entidades Financiadoras...), a seguir o doente é encaminhado às repetitivas salas de espera onde depois será chamado para a realização do ato medico previsto.

A realização do check-in é maioritariamente feita de forma presencial e com recurso aos funcionários administrativos das unidades. Esta tarefa é completamente determinística e modelável, de tal forma que poderia ser sujeita a uma automatização quase completa de forma a evitar o recurso a interação pessoal com administrativos deixando assim a necessidade desta interação só para tratar das exceções.

Depois do doente ser atendido pelos profissionais de saúde, é encaminhado para realização dos procedimentos de saída ou check-out onde se efetua o pagamento (no caso de haver valores por pagar) e a marcação de atos clínicos posteriores. Atualmente este é um processo ausente de qualquer nível de informatização nas unidades de saúde para além do recurso ao HIS. É o processo que mais tempo consome aos administrativos e aos doentes. É um processo suscetível de ser informatizado se se resolverem os obstáculos de "Colaboração" entre as diferentes entidades envolvidas no processo: unidades hospitalares, entidades pagadoras, entidades bancárias, doentes. O que implicaria que o cálculo do pagamento (caso exista) pudesse ser automático e todas as transações inerentes a este processo fazerem-se de forma totalmente informatizada.

Perspetiva Futura: Será inevitável num futuro próximo o aparecimento e desenvolvimento de apps que permitirão o acompanhamento do circuito dos doentes dentro de uma unidade de saúde. As principais funcionalidades que se espera virem a ser disponibilizadas por aplicativos informáticos para facilitar o percurso dos doentes nas suas visitas às unidades de saúde serão: a possibilidade dos

doentes realizarem a sua "pré-anamsese" antes da visita às unidades de saúde, a deteção automática de dispositivos, como smarphones, para registar a chegada dos doentes às unidades de saúde, a disponibilização de aplicativos GPS de edifícios que guiarão os doentes para as diferentes localizações dentro das unidades de saúde, o subministro de informação relevante para a gestão pessoal da estadia dos doentes nas unidades de saúde (como os tempos de espera previstos para cada ato clínico), a capacidade de estabelecer digitalmente a relação dos doentes com os seu seguros de saúde e as entidades bancárias para a realização do cálculo dos valores a pagar, a execução automática dos pagamentos e a capacidade de gerar propostas automáticas para as marcações das próximas visitas dos doentes às unidades de saúde no caso de serem necessárias. Este tipo de ferramentas beneficiará não só aos doentes como também as próprias unidades, assim como a todas as entidades vinculadas de forma direta ou indiretamente com a programação, a realização e o pagamento de atos clínicos.

4.2.3 Ato Clínico

Descrição: Olharemos neste ponto para o ato clínico desde uma perspetiva administrativa e não clínica, como já o fizemos anteriormente. Neste momento a realização de um ato clinico é quase na totalidade das vezes feito de maneira presencial. É um cara a cara entre os profissionais de saúde implicados na realização do ato clínico e o doente. Surgem de uma forma tímida alternativas a este modelo, baseadas num conceito, antigo de nome e recente em utilização real: a Telemedicina. Obviamente o bom senso leva-nos a pensar neste conceito onde existe viabilidade na sua aplicação, em cenários onde a presença física dos doentes e/ou profissionais de saúde não é absolutamente necessária para a realização do ato em questão.

Constrangimentos, não só clínicos, mas sobretudo de caráter regulamentar, logístico e financeiro tem impedido a evolução deste conceito. A não existência de legislação a este respeito faz com que as unidades de saúde e as entidades financiadoras de saúde evitem encarar este modelo de uma forma séria e determinada.

O HIS é a única ferramenta de cotidiana utilização no que a sistemas de informação se refere na realização de atos clínicos, utilizado principalmente para consultar e registar informação no registo de saúde eletrónico (EPR) do doente e para gerir informação administrativa e financeira referente à relação entre o doente, as entidades de saúde e as entidades financiadoras.

Perspetiva Futura: Se tomarmos como exemplo e referência o ato médico "consulta médica" e o desconstruímos tentando detetar as atividades mais importantes que nele acontecem, podemos destacar as seguintes: anamnese feita pelo doente, anamnese feita pelo profissional de saúde, solicitação de meios complementares de diagnóstico, envio e receção de meios complementares de diagnóstico, revisão por parte do profissional de saúde de meios complementares de diagnóstico, elaboração e envio de prescrições médicas e o ato de auscultação/revisão feita pelo profissional de saúde ao doente. Todas estas atividades, excetuando a auscultação/revisão do doente, podem ser sujeitas a mecanismos de automatização e virtualização, e realizadas à distância sem a necessidade da presença física do doente. É neste sentido que se prevê o caminho a seguir no que à

transformação digital diz respeito, na vertente de realização de atos clínicos dentro do contexto da informatização da saúde.

É um estândar que a duração de uma consulta medica é de aproximadamente 15 minutos, dos quais metade do tempo é dedicado à anamnese, ficando a outra metade para a realização do diagnóstico ou pedido de meios complementares de diagnóstico, revisão de resultados de meios complementares de diagnóstico, elaboração de prescrições e os respetivos rituais sociais de cumprimentos e despedidas. Podemos prever as vantagens em termos de eficiência que a automatização, virtualização e a remota ação deste processo pode trazer, sem pretender substituir o modelo do atual, mas sim oferecer uma alternativa para quem quiser tirar partido deste tipo de soluções [7].

Existem ferramentas que permitem a automatização deste processo e já há exemplos de medicina à distância exercida sobretudo para lugares de escassez de profissionais de saúde. Este conceito pode ser banalizado e ser disponibilizado para o uso cotidiano, evitando deslocações desnecessárias e otimizando o tempo empregue nestas atividades por todos os que nelas se encontram envolvidos.

Anamneses feitas pelos doentes antes de ir as consultas ("pré-anamnese"), envio e revisão de resultados de meios complementares de diagnóstico, segundas opiniões médicas, teleconsultas, são entre outras atividades para as quais se espera o surgimentos e evolução de ferramentas informáticas tipo apps ou sites ou mesmo dispositivos de hardware que permitam e/ou facilitem a execução à distância destas atividades.

5 Conclusões

Depois de ter feito este percurso pelos processos mais importantes integradores de uma unidade hospitalar, olhando para o seu atual estado no que a estratégia digital se refere, e as perspetivas futuras de desenvolvimento que cada um tem, podemos concluir o seguinte:

5.1 A saúde é uma área de foco para os investigadores e investidores protagonistas do I&D na sociedade atual por ser este um terreno fértil e ainda pouco explorado no âmbito da utilização de ferramentas informáticas no seu cotidiano [1]. A área dos equipamentos médicos, os medicamentos e algumas ferramentas de comunicação/interação com os utilizadores dos serviços de saúde, como por exemplo os sites e as apps, são mostra do exponencial crescimento que a área hospitalar tem tido no âmbito da utilização de ferramentas informáticas. Este novo paradigma tem sido um dos eixos fundamentais na alavancagem do seu desenvolvimento.

5.2 A utilização de ferramentas informáticas no âmbito do apoio aos profissionais de saúde na elaboração do diagnóstico e a definição da terapêutica, e de maneira geral, no âmbito do apoio à decisão clínica, está ainda num estado precário devido à falta de aceitação que estas ferramentas tem tido no seio da comunidade dos profissionais de saúde. Apesar dos investimentos feitos neste domínio e das tentativas de dinamização destas ferramentas, os resultados não têm sido concordantes com os esforços realizados.

5.3 As linhas modernas de investigação clínica centradas principalmente na Genómica e a Nanotecnologia estão abrindo caminho para novas áreas na saúde como a Medicina de Precisão ou Medicina Personalizada as quais estão suportadas no uso de ferramenta informáticas. O que trará

como consequência a aceitação das mesmas no seu cotidiano e consequentemente potenciarão a sua evolução.

5.4 O recurso a grandes volumes de dados clínicos em tempo real ou quase em tempo real com a possibilidade de os analisar exaustivamente e extrair deles novos conhecimento ou verificar hipóteses formuladas, abrirá caminho e acrescentará credibilidade às ferramentas de apoio à decisão clínica.

5.5 A virtualização de funcionalidades na área da gestão hospitalar será cada vez maior com o aparecimento e evolução de novas apps de ajuda à gestão preventiva da saúde, à gestão da convivência com as doenças crónicas, à monitorização remota dos doentes, ao seguimento de doentes no seu processo de recuperação fora das unidades de saúde e ao processo de avaliação mútua entre doentes e entidades, que tanta falta faz na promoção da excelência clínica (medição de outcomes clínicos).

5.6 Finalmente vale a pena analisar que, tendo em conta o desmembramento dos cuidados de saúde devido ao processo cada vez maior de especialização nas diferentes áreas da medicina e o aparecimento do conceito de Medicina do Retalho torna-se indispensável o repensar do modelo de negócio inerente a esta área, que se tem mantido igual desde sempre e que tem resistido a todo tipo de evolução: organizacional e tecnológica. O conceito de "Colaboração", que já conquistou terreno noutras disciplinas, tem uma função preponderante na mudança da área da saúde no seu processo de transformação digital. O olhar para um doente desde uma perspetiva total e não como um conjunto de partes, cada uma atendida por especialistas isolados (ou quase isolados), será um desafio a curto prazo. O modelo de organização no qual as unidades de saúde apresentam resultados de acordo com o número de atos clínicos isolados que realizam e calculam os indicadores de excelência clínica com base nos processos de avaliação das atividades intra-hospitalares, tende a mudar para um modelo onde os resultados serão apresentados com base na qualidade e rapidez de solução que ofereçam aos problemas de saúde dos doentes [5], e que só termina depois da recuperação "total" do doente mesmo aquela que acontece fora das unidades de saúde. O conceito de "Colaboração" traz consigo implícita a necessidade da definição de uma Semântica Clínica que permita a implementação de mecanismos de interoperabilidade tecnológica e facilite a comunicação entre todo o ecossistema da saúde [3]: doente, profissionais de saúde, unidades de saúde, entidades financiadoras de saúde, farmacêuticas e as entidades governamentais que fazem a gestão deste domínio.

Bibliografia

[1] Keasberry, J., Scott, I. A., Sullivan, C., Staib, A., & Ashby, R. (2017). Going digital: a narrative overview of the clinical and organisational impacts of eHealth technologies in hospital practice. Australian Health Review.

[2] Topol, E. (2013). The creative destruction of medicine: How the digital revolution will create better health care.

[3] Bigus, J. P., Campbell, M., Carmeli, B., Cefkin, M., Chang, H., Chen-Ritzo, C. H., ... & Glissmann, S. (2011). Information technology for healthcare transformation. IBM Journal of Research and Development, 55(5), 6-1.

[4] Paes, L. R. D. A. (2003). O uso da informática no processo de tomada de decisão médica em cardiologia: um estudo de casos múltiplos em hospitais de São Paulo.

[5] Ford, G., Compton, M., Millett, G., & Tzortzis, A. (2017). The Role of Digital Disruption in Healthcare Service Innovation. Service Business Model Innovation in Healthcare and Hospital Management, 57-70.

[6] Ghassemi, M., Celi, L. A., & Stone, D. J. (2015). State of the art review: the data revolution in critical care. Critical Care, 19(1), 118.

[7] Riva, G. (2000). From Telehealth to E-health: Internet and distributed virtual reality in health care. CyberPsychology & Behavior, 3(6), 989-998.

[8] Honeyman, M., Dunn, P., & McKenna, H. (2016). A digital NHS?.

[9] Hwang, J., & Christensen, C. M. (2008). Disruptive innovation in health care delivery: a framework for business-model innovation. Health Affairs, 27(5), 1329-1335.

[10] Muhos, M., Del Foit Jr, L. R., & Saarela, M. (2016, September). Growth Management of Digital Health Care Service Start-Ups–California Case Studies. In Proceedings of The 11th European Conference on Innovation and Entrepreneurship 15-16 September 2016 (p. 512).

Learning Occupancy Grid Maps Using Simulated Annealing

Hongjun Li, Miguel Barão and Luís Rato

Departamento de Informática,
Universidade de Évora, Portugal
li.hongjun@foxmail.com,{mjsb,lmr}@uevora.pt

Abstract. Solving the simultaneous localization and mapping problem is to extract the information about the robot pose and the map for the observed space. Based on the observations from the sensors, some information on the unexplored space can also be obtained. This paper focuses on how to obtain information on unknown parts from the previously observed space. Occupancy grid maps will be built for the environment. Markov random field model is applied to consider the dependence between grid cells. By maximizing the posterior distribution of Markov random field model, the information on unknown part can be obtained. Simulated annealing is used to solve the maximum problem.

Keywords: Occupancy grid maps, Markov random field, Simulated annealing

1 Introduction

Simultaneous localization and mapping (SLAM) is the problem that how to build maps and localize themselves simultaneously when robots explore and obtain observations in the unknown environment. This problem can be factored into a vehicle problem and a conditional mapping problem [1]. Except the basic task to know the robot pose and built a map for the observed space, a lot of further work for unknown space can be done based on the information the robots have obtained from the sensors.

The simplest task is exploration. The robots should know where to go to gain the most new information about the world based on what they have known. The basic strategy is frontier-based exploration [2]. The robots should detect the frontiers, which are the regions on the border between known space and unexplored space, and move to the nearest frontier. The next best view (NBV) is used to construct a polygonal layout of an environment for exploration [3]. The NBV is also applied to help the robot explore the unexplored space of the map [4] and this approach considers the interaction between sensor and obstacle.

The further work is to extract information on the unknown space. Based on the observation of the surroundings of an unexplored region, prediction-based SLAM algorithm (P-SLAM) is proposed to predict the structure inside an unexplored region[5]. Generalized Voronoi Graph (GVG) is applied to predict the

map structure [6], such as road, intersection and gateway. [7] develops a cell decomposition algorithm for path planning in an unknown environment. By finding similarities between the current surroundings of the robot and known environment, [8] proposes a novel approach to predict how the environment may expand in the unknown areas.

In this paper, the observations from range sensor are used to build an occupancy grid map for the environment. Markov random field (MRF) model is applied to represent the dependence between the grid cells and their neighbors. By maximizing the posterior distribution of MRF model, the unknown space can be predicted. Simulated annealing (SA) is applied to optimize the posterior distribution. In section 2, we assume the grid cells are independent and compute the probability of each grid cell recursively based on Bayes law. Each grid cell is labelled based on the corresponding probability and the MRF model of the map is built in Section 3. In Section 4, SA is applied to optimize the posterior distribution of MRF model. Finally, a small simulation is done in Section 5.

2 Observation

In this part, we do not consider the dependence between grid cells. We assume observations of grid cells are independent of each other and the current observation of the map is independent of the previous observations. In a grid map, each grid cell has two possible states: occupied and free. They are labelled as $X = \{1, -1\}$. This work will build a grid map with range finders. The robot can only observe a small part of the map once. If the grid cell is inside the measurement range, we can know the probability. When the cell is outside the measurement range, the value should be 0.5. As time goes, the robot gets a lot of observations. Based on Bayes law, we obtain

$$p(m|z^{1:t}) = \frac{p(z^t|m)p(m|z^{1:t-1})}{p(z^t)}, \tag{1}$$

where $m = \{m_0, m_1, ..., m_n\}$ is the set representing all the grid cells and $z^{1:t} = \{z^1, z^2, ..., z^t\}$ is the set representing all the observations. $p(z^t|m)$ is the measurement probability, $p(z^t)$ is the normalizer and $p(m|z^{1:t})$ is the probability distribution over the map, conditioned on all past measurements $z^{1:t}$. Since we have independence, we can do it for each cell individually as

$$p(m_i|z_i^{1:t}) = \frac{p(z_i^t|m_i)p(m_i|z_i^{1:t-1})}{p(z_i^t|z_i^{1:t-1})}, \tag{2}$$

where

$$p(z_i^t|z_i^{1:t-1}) = \sum_{m_i \in X} p(z_i^t|m_i)p(m_i|z_i^{1:t-1}). \tag{3}$$

Finally, every grid cell is labelled as -1, 0, 1 based on the probability $p(m_i|z_i^{1:t})$. If the probability of m_i is more than 0.5, the label L_i is 1 and the grid cell with

a probability of less than 0.5 is labelled as -1. If the probability is 0.5, the corresponding grid cell is labelled as 0. The labels are used as observations in the subsequent chapters.

3 The MRF Model

The map is regarded as a two dimension MRF and can be factorized according to the cliques of the graph. A clique c is defined as a subset of sites. Second-order neighborhood system, also called the 8-neighborhood system, is considered as shown in Figure 1. There are eight neighbors for every site and the clique consists of a single-site and a pair of neighboring sites as shown in Figure 2. The collections of single-site and pair-site cliques will be denoted by C_1 and C_2. The single-site cliques are not useful in this task, they are not considered in this paper.

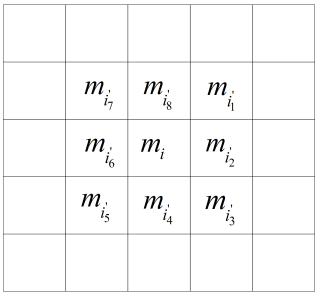


Fig. 1: Second-order neighborhood system

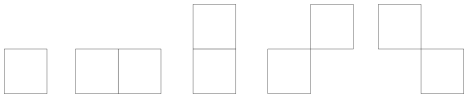


Fig. 2: Cliques in second-order neighborhood system

3.1 Prior Probability

The prior probability is formulated as

$$p(m) = \frac{1}{f} \mathrm{e}^{-\frac{1}{T}U(m)}, \tag{4}$$

where

$$f = \sum_{m \in \mathbb{M}} \mathrm{e}^{-\frac{1}{T}U(m)} \tag{5}$$

is the partition function [9], $\mathbb{M}$ represents all the configuration and T is a constant called the temperature. $U(m)$ is the energy function and formulated as

$$U(m) = \sum_{c \in C_2} V_c(m). \tag{6}$$

It is a sum of clique potentials $V_c(m)$ over all possible cliques C_2. The clique potential is defined as

$$V_c(m) = (m_i - m_{i'})^2, \tag{7}$$

where m_i and $m_{i'}$ are based on labels and they have two possible labels: -1 and 1.

3.2 Likelihood

The likelihood of m is

$$p(Z|m) = \frac{1}{\prod_i \sqrt{2\pi\sigma_i^2}} \mathrm{e}^{-U(Z|m)}, \tag{8}$$

where

$$U(Z|m) = \sum_i L_i^2 (m_i - L_i)^2 / [2\sigma_i^2]. \tag{9}$$

The unknown grid cells are not considered in the likelihood energy. When the grid cell is unknown, the label L_i is 0 and $L_i^2(m_i - L_i)^2$ is also 0. If the noise distribution is homogeneous, the deviations σ_i are the same as σ for all grid cells.

3.3 Posterior Probability

Based on Bayes rule, the posterior distribution is formulated as

$$p(m|Z) = \eta p(Z|m) p(m) \tag{10}$$

$$= \eta \frac{1}{\prod_i \sqrt{2\pi\sigma^2}} \frac{1}{f} \mathrm{e}^{-E(m)}, \tag{11}$$

where η is a constant and the posterior energy

$$E(m) = \sum_i L_i^2 (m_i - L_i)^2 / [2\sigma^2] + \sum_i \sum_{i'} (m_i - m_{i'})^2 \tag{12}$$

is the sum of likelihood energy $U(Z|m)$ and prior energy $U(m)$.

4 Simulated annealing

The predictive problem can be solved by maximizing the posterior distribution or equivalently minimizing the posterior energy. The posterior energy (12) can be changed to equation (13) equivalently.

$$E'(m) = \sum_i -m_i L_i/[2\sigma^2] + \sum_i \sum_{i'} -m_i m_{i'} \tag{13}$$

SA [10] is applied to solve the problem. The SA algorithm is described as following steps.

(1)Initially, T is set very high, m is set to the labelled observation map and the unobserved grid cells are labelled -1 or 1 randomly. The initial energy $E_0(m)$ is calculated by equation (13).

(2)Change the label of one grid cell m_i, calculate the new energy $E_{new}(m) = E_{old}(m) - E_{old}(m_i) + E_{new}(m_i)$.

(3)If $E_{new}(m) < E_{old}(m)$, the new state is accepted. If not, the new state is accepted by the acceptance probability

$$P(E_{new}(m), E_{old}(m), T) = \exp(\frac{E_{old}(m) - E_{new}(m)}{T}) \tag{14}$$

If the acceptance probability is more than a random probability, the new state is also accepted. Otherwise, it is rejected.

(4)If all the grid cells are searched in order, go to next step. Otherwise, return to step (2).

(5)If $T \to 0$, return m. Otherwise, T is decreased according to the following schedule

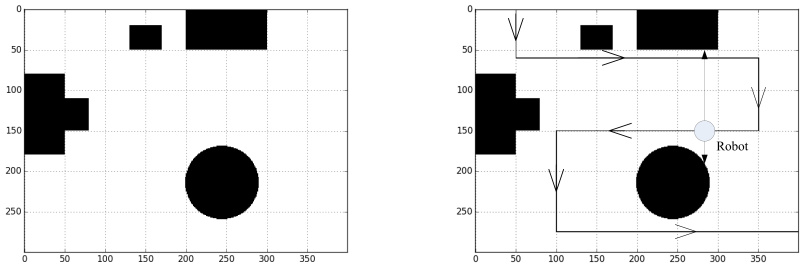
$$T^{(t)} = kT^{(t-1)} \tag{15}$$

and return step (2).

5 Simulation

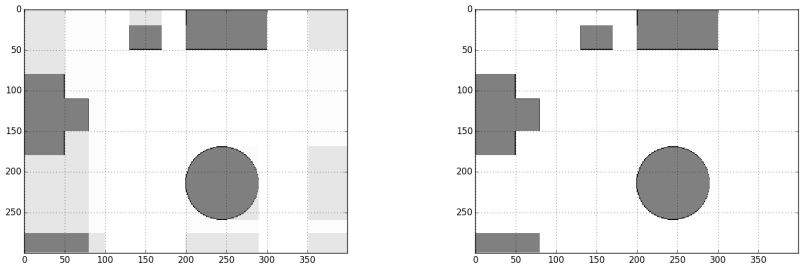
The true map is shown as Figure 3(a). There are some objects in the map. They have different shapes and sizes. The robot moves from the top to the right side, the trajectory is shown as Figure 3(b). At a position, there are two measurement directions: $\pm\pi/2$. They are relative to the robot direction.

Laser sensor is used to get information from the environment. The beam of the laser sensor has the same width with the grid cell and maximum range is 400 grid cells. Following along a line in the measurement direction the cells are occupied with low probability 0.1, at least until the measured distance. At the distance the cell is occupied with high probability 0.9. The observation is shown as Figure 4(a). Only some borders of the objects are observed occupied. All the free space except left-bottom corner is observed free. Other parts are not observed. The labels are shown as Figure 4(b).



(a) (b)

Fig. 3: True map and the trajectory



(a) (b)

Fig. 4: Observations

Initially, $\sigma^2 = 0.25$, $T = 50$. After optimized by SA, the simulation result is shown as Figure 5. Based on the observed borders, this method is trying to predict the shape of the objects. Because of the lack of some borders, it can not construct the objects precisely.

6 Conclusion

In this paper, Bayes law is used to compute the observations for each grid recursively. Based on the observations, MRF is applied to build the model of the occupancy grid maps. Omitting the unknown space in likelihood energy, the unknown parts can be inferred from the observations. SA algorithm is used to minimize the posterior energy. In future work, we will explore the dynamic environment based on this work.

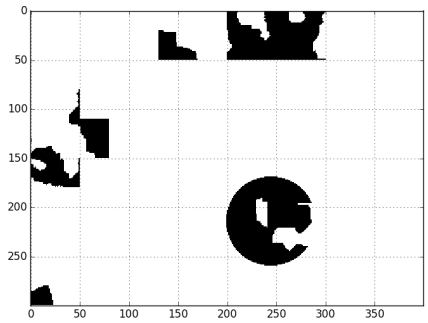


Fig. 5: Predictive map

Acknowledgment

This work was supported by EACEA under the Erasmus Mundus Action 2, Strand 1 project LEADER - Links in Europe and Asia for engineering, eDucation, Enterprise and Research exchanges.

References

1. Hugh Durrant-Whyte and Tim Bailey. Simultaneous localization and mapping (slam): Part i. *IEEE Robotics and Automation Magazine*, 13(2):99–110, 2006.
2. Brian Yamauchi. A frontier-based approach for autonomous exploration. In *Proceedings of the IEEE International Symposium on Computational Intelligence in Robotics and Automation*, pages 146–151. IEEE, 1997.
3. Héctor González-Baños, Eric Mao, Jean-Claude Latombe, TM Murali, and Alon Efrat. Planning robot motion strategies for efficient model construction. In *Proceedings of the International Symposium on Robotics Research*, volume 9, pages 345–352, 2000.
4. Robert Grabowski, Pradeep Khosla, and Howie Choset. Autonomous exploration via regions of interest. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 2, pages 1691–1696. IEEE, 2003.
5. H Jacky Chang, CS George Lee, Yung-Hsiang Lu, and Y Charlie Hu. P-slam: Simultaneous localization and mapping with environmental-structure prediction. *IEEE Transactions on Robotics*, 23(2):281–293, 2007.
6. Shu Yun Chung and Han Pang Huang. Simultaneous topological map prediction and moving object trajectory prediction in unknown environments. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1594–1599. IEEE, 2008.
7. Batsaikhan Dugarjav, Soon-Geul Lee, Donghan Kim, Jong Hyeong Kim, and Nak Young Chong. Scan matching online cell decomposition for coverage path planning in an unknown environment. *International journal of precision engineering and manufacturing*, 14(9):1551–1558, 2013.

8. Daniel Perea Ström, Fabrizio Nenci, and Cyrill Stachniss. Predictive exploration considering previously mapped environments. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 2761–2766. IEEE, 2015.
9. Stan Z Li. *Markov random field modeling in image analysis*. Springer Science & Business Media, 2009.
10. Peter JM Van Laarhoven and Emile HL Aarts. Simulated annealing. In *Simulated Annealing: Theory and Applications*, pages 7–15. Springer, 1987.

Learning Analysis Using Natural Language Processing

Rodwan Bakkar Deyab, Irene Rodrigues

Universidade de Évora, Department of Informatics,
Rua Romão Ramalho n59, 7000-671 Évora, Portugal
d34642@alunos.uevora.pt, ipr@uevora.pt

Abstract. In this work, we present a Question Answering System to make analysis about courses of Learning Management Systems. The Learning Management System depends on an ontology. In this work we use data from the University of Evora Moodle for the ontology. This system uses Natural Language Processing to convert the natural language question to SPARQL query which is used to query the ontology. The SPARQL query result is presented graphically to the user. This graphical representation gives a general view about the content of the Learning Management System and makes it easier to draw conclusions about it.

Keywords: Learning Management Systems, Ontology, Natural Language Processing, Question Answering Systems

1 Introduction

Learning Management Systems are applications which facilitate some tasks in e-Learning like documentation, reporting, etc. It is possible to benefit the data these systems have in their databases to create and populate ontologies. Ontologies structure hold meaning. It makes it possible to create Question Answering Systems upon. The question answering systems work on converting natural language questions to queries, like SPARQL[1], which can be run on the ontologies. The results of these queries can be presented graphically to the user. This paper is organized as following:

Section2: presents Ontologies and Learning Management Systems. Section3: presents Natural Language Processing as a tool for Question Answering Systems. Section4: presents Question Answering Systems and the algorithm we proposed to convert natural language questions to SPARQL queries. Section5: presents some examples of natural language questions and it graphically presents the results of these questions after running on the ontology. Section6: presents conclusions and future work.

[1] https://www.w3.org/TR/rdf-sparql-query/

2 Ontologies and Learning Management Systems

In philosophy, *"ontology is the part which is concerned in understanding the nature of existence"*.

In computer science, ontology is defined as[3]:

"A specification of a representational vocabulary for a shared domain of discourse definitions of classes, relations, functions, and other objects is called an ontology"

And formally [9]:

A (core) ontology is a tuple $\Omega := (C, is\_a, R, \sigma)$ where C is a set whose elements are called *concepts*, is_a is a partial order on C (i.e., a binary relation $is\_a \subseteq C \times C$ which is reflexive, transitive, and anti-symmetric), R is a set whose elements are called *relation names* (or *relations* for short), and $\sigma : R \to C^+$ is a function which assigns to each relation name its arity.

In this work we consider the Moodle Learning Management System[2]. An ontology was built manually using Protégé[3]. This ontology was populated using data from the University of Evora Moodle database. Ontology Population [7] was done using Information Extraction [11]. Figure 1 presents a part of the ontology structure using the OntoGraf[4] utility of protégé.

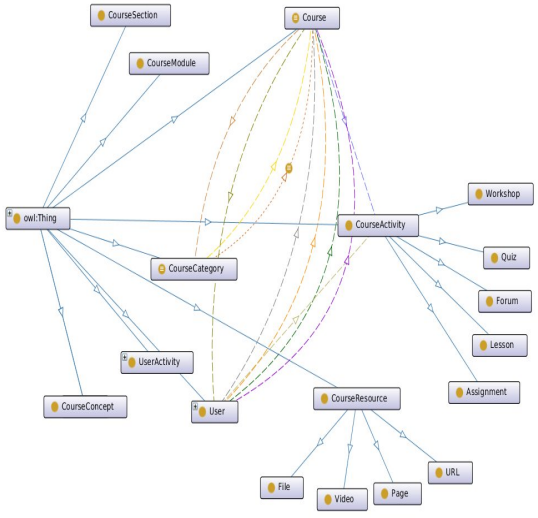


Fig. 1: The Ontology View, OntoGraf

[2] https://moodle.com/
[3] http://protege.stanford.edu/
[4] http://protegewiki.stanford.edu/wiki/OntoGraf

3 Natural Language Processing

Natural Language Processing is an area of research considered as a subfield of Artificial Intelligence. This area is concerned about making the computer understand natural language text and its meaning. It is an interdisciplinary such that it depends on many disciplines like machine learning, linguistics, mathematics, artificial intelligence, information science and psychology. In order to achieve its tasks, understanding of phonetics, morphology, grammar, lexicology and semantics should be present.

Natural Language Processing is used in Question Answering Systems to convert question to a query language, SPARQL, for example. This will be presented in the next section. Some of the NLP tasks are:

3.1 Part-of-Speech(POS) Tagging

It is assigning the syntactical part of speech for each word in a sentence. For example, it tags a word as a verb, noun, adjective, etc. This task is useful for other tasks like Named Entity Extraction(NER). POS Tagging can be achieved by many approaches[5]:

- Supervised Taggers:
  - Rule-Based: Brill Tagger
  - Stochastic: Hidden Markov Model(HMM)
  - Neural Network
- Unsupervised Taggers:
  - Rule-Based: Brill Tagger
  - Transformation-Based: User Baum-welch
  - Neural Network

In this work, Stanford POS-tagger[10] was used.

3.2 Named Entity Recognition NER

Named entity refers to an entity like place, organization, person, date, etc. NER works on defining the named entities in the textual context. It defines the boundaries on the named entity (some entities can be more than one word like "New York", for example).

Machine Learning is used to achieve NER. There are three approaches: supervised learning, semi-supervised learning and unsupervised learning.

3.3 Dependency Parsing

Dependency parsing determines the syntactic relations between words in the sentence. It resolves ambiguity. For example, an ambiguous sentence "I saw my friend with glasses". This sentence has two possible meanings:

- I saw my friend, I was using glasses.

– I saw my friend, he was using glasses.

Using the Stanford Dependency Parser Online Demo[5], figure 2 presents the dependency parsing of the sentence. This dependency parsing shows that the sentence has the second meaning removing the ambiguity.

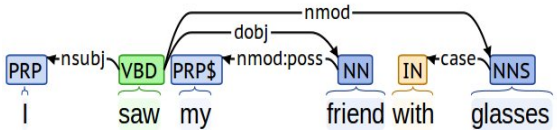


Fig. 2: Dependency parsing example

Some dependency parsing approaches are:

– Shift-reduce [8].
– Spanning tree [6].
– Cascaded chunking [4].

In this work, the Stanford Dependency Parser [1] was used.

4 Question Answering System

In our Question Answering System, natural language questions are processed using the natural language tools in GATE [2]. GATE is a natural language processing framework that includes tools to produce annotations for the sentence tokens. Stanford POS tagger and stanford dependency parser are included in this framework as tools. Figure 3 shows the processing pipeline, each process is a tool included in the GATE framework. Consider question (1):

Question(1): "What is the total number of activities for each student in the course 1545".

This sentence will pass through the pipeline process producing a set of annotations. The steps of this pipeline are (see figure 3):

– Document reset: clears all the annotations in the sentence.
– Regular expression sentence splitter: it splits the text into sentences.
– Stanford PTB tokenizer: it annotates tokens with the following information:
  • Type: a token
  • Start: the index of the first character of the word.
  • End: the index of the last character of the word.

[5] http://nlp.stanford.edu:8080/corenlp/

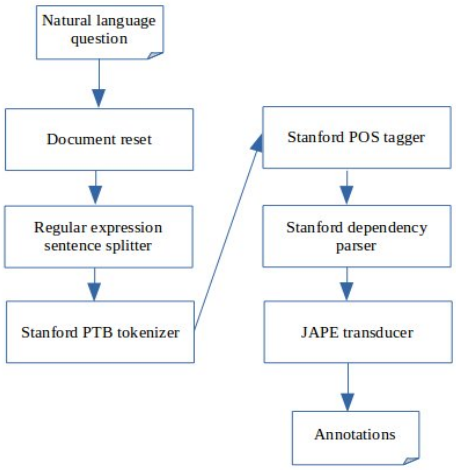


Fig. 3: Processing pipeline

- Id: the identifier of the token
- Features: some attributes of the word such as: kind, length, orth and string.

For example, the word "What" is tokenized as: type=Token, start=0, end= 4, Id=4, Features={kind=word, length=4, orth=upperInitial, string=What}.

– Stanford POS tagger: it will add the feature "category" to the token. For example, the token 4 will have the feature category=WP which means "Wh-pronoun".
– Stanford dependency parser: it will add the feature "dependencies" to the token. For example, the token 13 will have the feature dependencies=[pobj(15)] which means that the token 15 is a prepositional object of the token 13.
– JAPE transducer: JAPE rules to extract the annotations from the GATE framework and store them in a file. JAPE (Java Annotation Patterns Engine) instructions are regular expressions on annotations.

Figure 4 presents some of the annotations generated for Question(1), in figure 5 we present a graphical representation of these annotations[6].

The algorithm proposed for the conversion from natural language to SPARQL:

4.1 Conversion Algorithm

An algorithm was created to convert the generated annotations from the previous step to a SPARQL query. The algorithm is presented in figure 6.

In the next subsections we explain the steps to interpret Question(1) as an example.

[6] This graph was obtained using Stanford CoreNLP Online Demo

```
AnnotationImpl: id=4; type=Token;
features={string=What, category=WP, dependencies=[cop(5), nsubj(11)]}

AnnotationImpl: id=5; type=Token; features={string=is, category=VBZ}

AnnotationImpl: id=7; type=Token; features={string=the, category=DT}

AnnotationImpl: id=9; type=Token; features={string=total, category=JJ}

AnnotationImpl: id=11; type=Token;
features={string=number, category=NN, dependencies=[det(7), amod(9), prep(13), nmod(15)]}

AnnotationImpl: id=13; type=Token;
features={string=of, category=IN, dependencies=[pobj(15)]}
...
```

Fig. 4: annotations for Question (1)

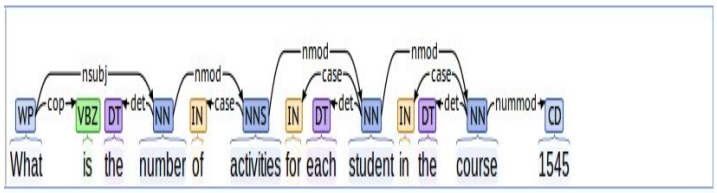


Fig. 5: POS-tags and dependencies of Question(1)

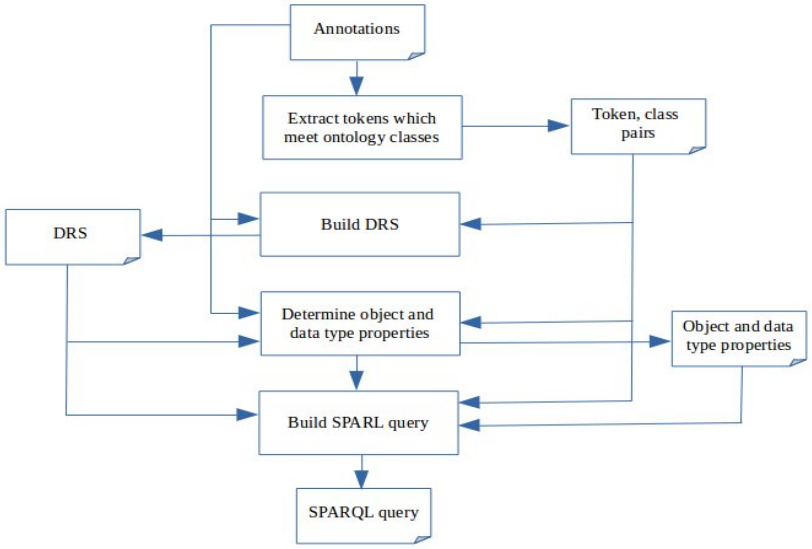


Fig. 6: The SPARQL-query-builder algorithm

Discourse entities Discourse entities are recognized in the process "Extract tokens which meet ontology classes". The algorithm will check all the nouns in the sentence. All the tokens marked as "NN" or "NNS" are nouns. The discourse are candidates to match ontology classes. The tokens marked as "NNS" are in plural form so they were lemmatized to get the singular form. Then they are searched for in the ontology classes to find a match. For Question(1), the result is the following discourse entities:

- (A, 4, What, Wh).
- (X, 21, student, each)
- (Y, 27, course, the)
- (Z, 15, activities, -)
- (W, 11, number, the)
- (B, 29, 1545, -)

Then this module will try to match each discourse entity with ontology classes like following:

- The "student" token meets the "User" class: (X, 21, student, User)
- The "course" token meets the "Course" class: (Y, 27, course, Course)
- The "activities" token meets the "UserActivity" or the "CourseActivity" classes: (Z, 15, activities, [UserActivities, CourseActivities])
- the "number" token meets no class: (W, 11, number, null)

DRS conditions The discourse representation conditions are obtained by extracting the dependencies of prepositional phrases, verbs and adjectives.

The algorithm loops through all the annotations to extract conditions. For each dependency in the tokens features it will get the head and dependent of it. The discourse conditions for Question(1) are:

- of(number-11, activities-15) -> of(W, Z)
- for(activities-15, student-21) -> for(Z, X)
- in(student-21, course-27) -> in(X, Y)
- is(What-4, number-11) -> is(A, W)
- total(number-11) -> total(W)
- num(course,1545) -> num(Y,B)

These conditions will be interpreted in the ontology in order to be translated into ontology properties.

Determine object and data type properties The DRS will be used to determine the object and data type properties from the ontology.

- for(Z, X) will produce the "activityByUser" object property. It will be obtained by searching for the object properties between the two classes. This is achieved using Jena reasoner.
- in(X, Y) will produce the "isStudentOf" object property.
- of(W, Z) will produce the "num" data type property.
- num(Y, B) will produce the "c_id" data type property.

Generating SPARQL query A SPARQL query is built using the DRS.

1. each discourse entity that was matched with a class will give rise to a variable definition in SPARQL query.
   (1), (2), (3) were generated by the tokens X, Y, Z respectively.
2. each object property or data type property will generate a constraint.
   (4), (5), (6) were generated using the object properties.
   (7) was generated using the data type property.
   (8) was generated using the num(Y, B) from DRS.
3. the select statement in SPARQL is built by collecting all the discourse variables with the "each" quantifier followed by variables that have "Wh" as a quantifier. So it was generated using total(W), each(X).
4. The variables which have the quantifier "each" will be presented in the group by clause. Group by was generated using each(X).

Figure 7 presents the SPARQL query generated as a result.

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX ns: <http://www.semanticweb.org/courses-ontology-57#>
 select ?x (sum(?num) as ?numAct)
  where {
         ?x rdf:type ns:User.           (1)
         ?y rdf:type ns:Course.         (2)
         ?z rdf:type ns:UserActivity.   (3)
         ?z ns:activityInCourse ?y.     (4)
         ?x ns:isStudentOf ?y.          (5)
         ?z ns:activityByUser ?x.       (6)
         ?z ns:num ?num.                (7)
         ?y ns:c_id '1545'.             (8)
        } group by ?x
```

Fig. 7: the activities of students in course 1545

This SPARQL query will be run on the ontology and the result will be presented graphically. This will be presented in the next Section.

5 Graphical Examples

We give some examples of some questions and the equivalent SPARQL queries for them. Then we present, graphically, the results of running these SPARQL queries on the ontology. JFreeChart[7] is used for the graphical presentation.

Figure 8 shows the result of the SPARQL query in figure 7.

Consider question:

Question(2): "I want to know the correlation between the number of activities each user does in course 1545 and his grade in this course".

Figure 9 presents the SPARQL query generated as a result.

[7] http://www.jfree.org/jfreechart/

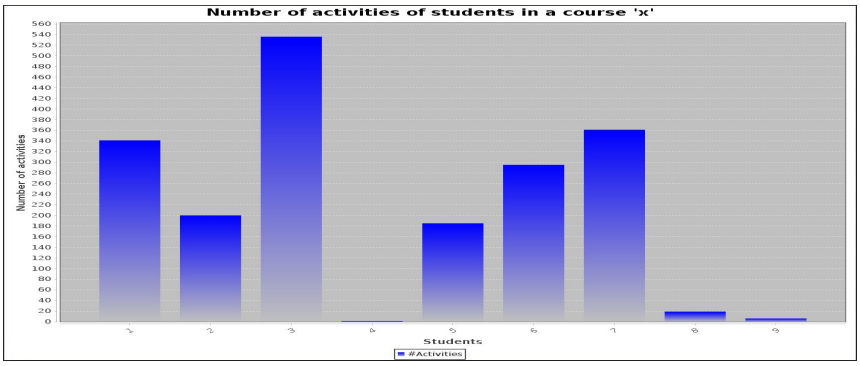


Fig. 8: Students activities in course 1545

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX ns: <http://www.semanticweb.org/courses-ontology-57#>
 select ?user ?gValue ?gPassed (sum(?num) as ?numAct)
  where {
         ?user rdf:type ns:User.
         ?course rdf:type ns:Course.
         ?course ns:c_id '1545'.
         ?userActivity rdf:type ns:UserActivity.
         ?grade rdf:type ns:Grade.
         ?userActivity ns:activityInCourse ?course.
         ?user ns:isStudentOf ?course.
         ?userActivity ns:activityByUser ?user.
         ?userActivity ns:num ?num.
         ?grade ns:gradeObtainedBy ?user.
         ?grade ns:gradeInCourse ?course.
         ?grade ns:gradeValue ?gValue.
         ?grade ns:gradePassed ?gPassed.
        } group by ?user ?gValue ?gPassed
          order by asc(UCASE(str(?user)))
```

Fig. 9: Correlation(activities-grades) in course 1545

Figure 10 presents the result of SPARQL query in figure 9. It has two scales, first one is for the number of activities and second one is for the grades.

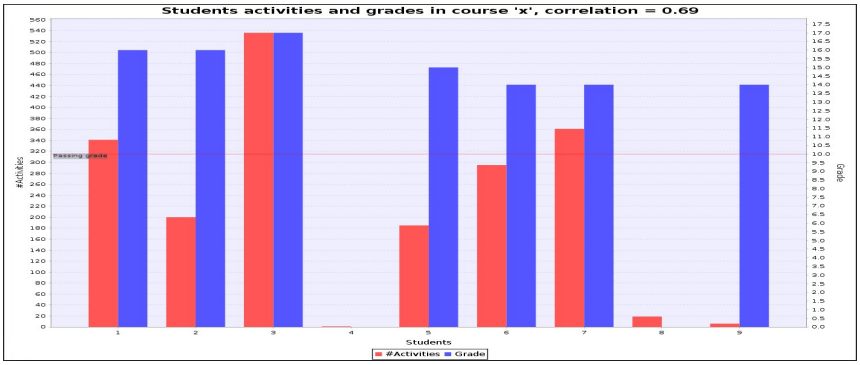


Fig. 10: The correlation (activities, grades) of students in course 1545

As the correlation has a positive value, we infer that the two variables (number of activities, grade) are positively related which means that students with more activities have better grades.

Consider question:

Question(3): "I want to know the activities of the student by week and their grades in course 1545".

Figure 11 presents the SPARQL query generated as a result.

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX ns: <http://www.semanticweb.org/courses-ontology-57#>
 select ?user ?week ?grade ?gValue ?gPassed (sum(?num) as ?numActWeek)
  where {
         ?user rdf:type ns:User.
         ?course rdf:type ns:Course.
         ?userActivity rdf:type ns:UserActivity.
         ?userActivity ns:activityInCourse ?course.
         ?user ns:isStudentOf ?course.
         ?userActivity ns:activityByUser ?user.
         ?userActivity ns:week ?week.
         ?userActivity ns:num ?num.
         ?course ns:c_id '1545'.
         ?grade rdf:type ns:Grade.
         ?grade ns:gradeObtainedBy ?user.
         ?grade ns:gradeInCourse ?course.
         ?grade ns:gradeValue ?gValue.
         ?grade ns:gradePassed ?gPassed.
        } group by ?user ?week ?grade ?gValue ?gPassed
          order by asc(UCASE(str(?user)))
```

Fig. 11: The activities by week of students in course 1545

Figure 12 presents the result of this query. It has two scales: first scale is for the number of activities and second one is for the grade. The horizontal axis represents the students of the course. It is observed that students tend to have more activities in the first weeks than the last weeks.

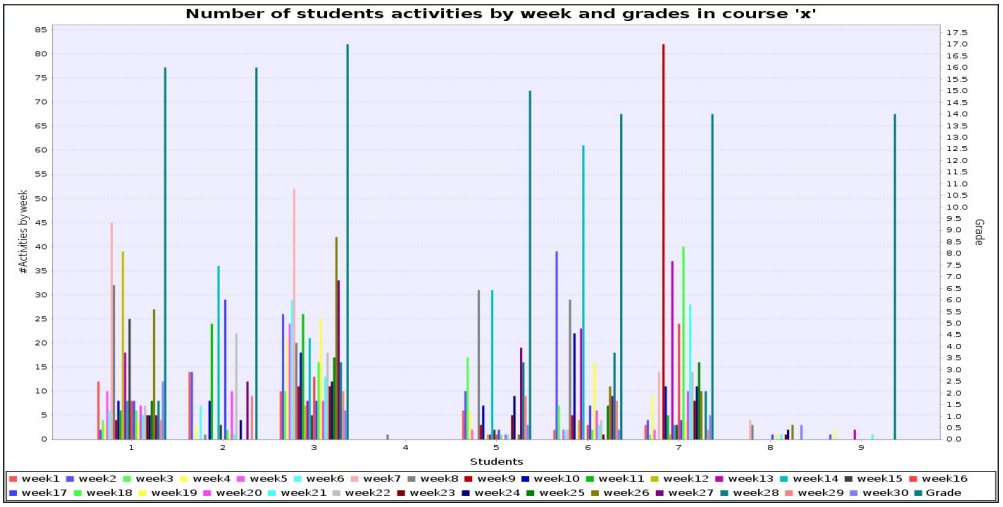


Fig. 12: The activities by week of the students in course 1545

6 Conclusion and Future Work

It was possible to present, graphically, information about the content of the Learning Management System which depends on an ontology. It was also possible to draw conclusions out of the graphical representations. This gives a general view about the information included in the Learning Management System.

As a future work, we can improve the Question Answering System to answer a wider range of natural language questions.

Acknowledgement

All sincere thanks to my professor Irene Rodrigues for her massive help. All thanks for my friend Fernanda Rosário for her support.

References

1. Danqi Chen and Christopher D Manning. A fast and accurate dependency parser using neural networks. In *EMNLP*, pages 740–750, 2014.

2. Hamish Cunningham. Gate, a general architecture for text engineering. *Computers and the Humanities*, 36(2):223–254, 2002.
3. Thomas R Gruber. A translation approach to portable ontology specifications. *Knowledge acquisition*, 5(2):199–220, 1993.
4. Taku Kudo and Yuji Matsumoto. Japanese dependency analysis using cascaded chunking. In *proceedings of the 6th conference on Natural language learning-Volume 20*, pages 1–7. Association for Computational Linguistics, 2002.
5. Deepika Kumawat and Vinesh Jain. Pos tagging approaches: A comparison. *International Journal of Computer Applications*, 118(6), 2015.
6. Ryan McDonald, Fernando Pereira, Kiril Ribarov, and Jan Hajič. Non-projective dependency parsing using spanning tree algorithms. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 523–530. Association for Computational Linguistics, 2005.
7. Georgios Petasis, Vangelis Karkaletsis, Georgios Paliouras, Anastasia Krithara, and Elias Zavitsanos. Ontology population and enrichment: State of the art. In *Knowledge-driven multimedia information extraction and ontology evolution*, pages 134–166. Springer-Verlag, 2011.
8. Kenji Sagae and Jun'ichi Tsujii. Shift-reduce dependency dag parsing. In *Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1*, pages 753–760. Association for Computational Linguistics, 2008.
9. Gerd Stumme and Alexander Maedche. Fca-merge: Bottom-up merging of ontologies. In *IJCAI*, volume 1, pages 225–230, 2001.
10. Kristina Toutanova and Christopher D Manning. Enriching the knowledge sources used in a maximum entropy part-of-speech tagger. In *Proceedings of the 2000 Joint SIGDAT conference on Empirical methods in natural language processing and very large corpora: held in conjunction with the 38th Annual Meeting of the Association for Computational Linguistics-Volume 13*, pages 63–70. Association for Computational Linguistics, 2000.
11. Daya C Wimalasuriya and Dejing Dou. Ontology-based information extraction: An introduction and a survey of current approaches. *Journal of Information Science*, 2010.

Query Expansion Techniques in Consumer Health Information Search *

Hua Yang

huayangchn@gmail.com
Informatics Department, University of Evora

Abstract. In this paper, state of the art query expansion techniques applied in health information search are introduced. Widely used techniques include thesaurus based query expansion and Pseudo Relevance Feedback techniques. Beside, we also explore using Word2vec models as another way to expand the original query. We experiment on different query expansion techniques and their combinations, results among different techniques are compared. Our experiment results show that Pseudo Relevance Feedback techniques improve retrieval performance in all cases. And a combined method of expansion with Word2vec model and Pseudo Relevance Feedback outperforms other methods in our experiment.

Keywords: query expansion, health information search, UMLS, Word2vec

1 Introduction

For consumer health information retrieval, the lay queries proposed by consumers (lay users) can prevent them from finding relevant contents written in more professional medical language. Vocabulary gap exists between consumers and medical experts. In dealing with this difficulty, abundant of work have been proposed to use query expansion techniques to expand the original query proposed by consumers, and have showed the effectiveness.

In this paper, we present the experiment of using different query expansion techniques. We use Pseudo Relevance Feedback techniques and domain specific thesaurus based expansion; we also experiment on a trained Word2vec model with Wikipedia. We establish a baseline and then compare the performance between the baseline and other techniques applied in our experiments. In the following of the paper, we first present related techniques used in our experiment in Section 2. Then we discuss the details of our experiments and presents the results on different techniques respectively in section 3. Conclusion is made in section 4. Finally, we propose our future work and research.

* This paper is for the assessment of Seminar 1.

2 Related work

Query expansion means expanding original query with extra terms. Query expansion is often an effective way to retrieve more relevant contents (Christopher and Hinrich, 1999; Carpineto and Romano, 2012). Regarding to the domain specific health information retrieval, early in 1996, a paper (Srinivasan, 1996) has pointed out that the best results obtained with expanded queries has a significant improvement over the unexpanded queries. A more recent survey (Palotti et al., 2015) on CLEF eHealth task also concludes that query expansion techniques play an important role in improving search effectiveness. Domain specific thesaurus UMLS[1] including varieties of meta-thesaurus like MeSH [2], CHV [3] has been used to extract synonyms or related terms for the terms in the original query. Controlled vocabularies in medical domain like MeSH can obviously improve the effectiveness of the retrieval system (Aronson and Rindflesch, 1997; Zhu and Carterette, 2012). CHV can be used to map technical terms to consumer friendly language and is effective in consumer health information search (Lopes and Ribeiro, 2016). Besides using thesaurus to expand the original query, using Word2vec models to find close words provides another way for query expansion. Word2vec model (Mikolov et al., 2013) is becoming one of the most efficient approach to learn word embeddings, since the learned representations in the model can be used to find the closest words for a user-specified word. A recent paper (Wang et al., 2015) points out that extracting close words from Word2vec models outperforms the state-of-the-art approaches on medical synonym extraction by a large margin.

2.1 Domain specific thesaurus

United Medical Language System (UMLS). For specific applications, medical information is characterized by a large diversity of vocabularies. Varieties of ways are used to express a same concept in different vocabularies. It is not easy to distribute useful information among different application systems. The lack of a common language has barred the interoperability of the applications in health area, United Medical Language System has been proposed to tackle the problem. UMLS aims to reduce barriers of computer applications to the health area and more specifically to the effective retrieval of machine readable information (Humphreys et al., 1998). UMLS contains three knowledge sources, Metathesaurus, Semantic Network, Specialist Lexicon and Tools. These knowledge sources can be used for information retrieval, natural language processing, automated indexing, thesaurus construction, electronic health records and others.

UMLS Metathesaurus clusters terms into concepts and assigns unique identifier to each concept. It contains biomedical and health related concepts, their

[1] https://www.nlm.nih.gov/research/umls/
[2] https://www.nlm.nih.gov/mesh/
[3] https://www.nlm.nih.gov/research/umls/CHV/index.html

various names and the conceptual relationship among its source vocabularies. Metathesaurus is not built in one vocabulary. Synonymous terms are clustered into a concept with a unique identifier (CUI). Each term is identified by a unique identifier (LUI). Each term is a normalized name and may have several strings (SUI), which represent the terms lexical variants in the source vocabularies. Each string is associated with one or more atoms (AUI) that represent the concept name. Figure 1 gives an example [4]. For a disease Headache, we can use UMLS to find its synonyms and related terms like *Cranial Pain* and *Head Pain Cephalgia* from different domain-specific vocabularies.

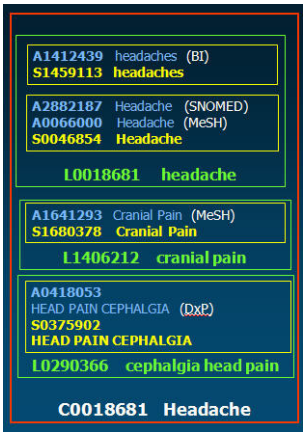


Fig. 1. An example of unique identifiers in UMLS

In TREC 2012 medical records track task, a paper (Voorhees and Hersh, 2012) points out that top performing groups each used some sort of vocabulary normalization device specific to the medical domain, supporting the hypothesis that language use within electronic health records is sufficiently different from general use to warrant domain specific processing. And they also point out that such devices must be used carefully as multiple groups also demonstrate that aggressive use harms baseline performance. In paper (Lu et al., 2009), the authors compare the different work of QE in the domain of biomedical text retrieval and point out that the results have been mixed. They pointed out that some papers show that the techniques could result in improved retrieval performance, while other papers show contradictory reports of using query expansion. Some work also show that using the concepts is not always effective, as presented in previous work (Darmoni et al., 2012) (Shen and Nie, 2015). A paper proposed that only the concepts that belong to some specific kinds of semantic type can be included in the expanded query (Balaneshin-kordan et al., 2015).

[4] Figure 1 is from https://www.nlm.nih.gov/research/umls

Consumer Health Vocabulary (CHV). Consumer Health Vocabulary translate technical terms to consumer friendly language. CHV connects informal, common words and phrases about health to technical terms used by health care professionals. It includes jargon, slang, ambiguous, and misspelled words used by consumers.

Medical Subject Headings (MeSH). The Medical Subject Headings thesaurus is a controlled vocabulary by National Library of Medicine (NLM). MeSH has three basic types of records: Descriptors, Qualifiers, and Supplementary Concept Records (SCRs). Each MeSH record has a preferred concept and in turn each concept has a preferred term (the name of the concept).

In a MeSH record, synonymous terms are grouped in a Concept. A Decriptor record consists of one or more concepts closely related to each other in meaning. Possible relationships between concepts are preferred term, reltated, narrower and boader, see figure 2 for more information (Darmoni et al., 2012). For example, the concept *Abortion Induced* has several other entry terms include *Abortion*, *Induced*, *Induced Abortion*. It also has some concept broader than itself like *Fertility Control*; narrower concepts like *Abortion Saline-Solution*; and related concepts like *Abortion Rate*. Narrower than relationship with one MesH Descriptor is the most common one in MeSH compared with broader than or related relationship (Darmoni et al., 2012).

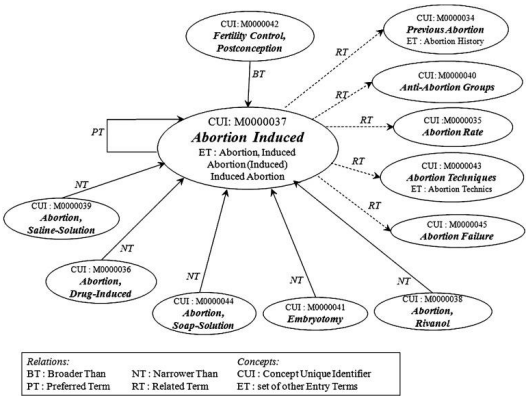


Fig. 2. Concepts Relationship in MeSH

2.2 Pseudo relevance feedback

Pseudo relevance feedback (a.k.a. blind relevance feedback) is a way to improve retrieval performance without the user interaction (Christopher and Hinrich, 1999). Previous works showed its effectiveness in improving the performance

(Voorhees et al., 2005) (Song et al., 2015). Figure 3 well explained how this technique can be used in an IR model to satisfy the user more [5]. Given a query q and the dataset, the retrieval system retrieved the dataset and returned an initial ranked list. Expanded terms are extracted from the top n documents in the initial list. An expanded query $q^{'}$ is generated, which includes the original query and the expanded terms. The search system retrieve the same dataset with the new generated query $q^{'}$ and and expansion-based list is produced.

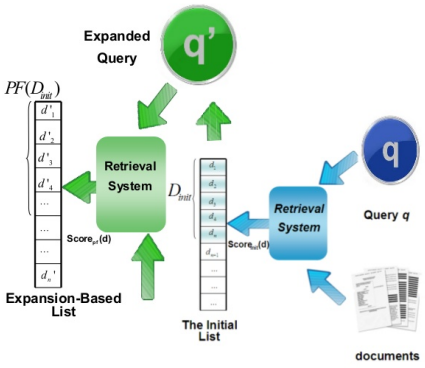


Fig. 3. Pseudo relevance feedback

2.3 Word2vec models

Word2vec can be used to train word embeddings (Mikolov et al., 2013). The vector representations of words learned by word2vec models have been shown to carry semantic meanings. Word2vec model is a shallow, two layer neural network that are trained to reconstruct linguistic contexts of words. Word2vec model takes as its input a large corpus of text and produces a high dimensional space (typically of several hundred dimensions), with each unique word in the corpus being assigned a corresponding vector in the space. Word vectors are positioned in the vector space such that words that share common contexts in the corpus are located in close proximity to one another in the space.

Continuous bag of word model (CBOW). CBOW model predicts one target word by the given context words. CBOW model takes the average of the vectors of the input context words, and uses the product of the input hidden weight matrix and the average vector as the output.

Skip Gram Model (SG). SG model is the opposite of the CBOW model, where the context words are predicted given a target word. The target word is at the input layer, and the context words are on the output layer.

[5] Figure 3 is from http://www.slideshare.net/

3 Experiments

In this section, we present our experiment on different techniques mentioned above. We first talk about the dataset used in our experiments. We build a baseline and then compare the performance among the baseline and the results on other techniques.

3.1 Data Corpus

We use the corpus from FIRE 2016 CHIS[6] task. This task includes 5 consumer health queries and corresponding dataset. The five queries proposed in the task are showed in table 1. The statistics of released training dataset is presented in table 2.

Table 1. 2016 FIRE CHIS queries

Query1: Dos sun exposure cause skin cancer
Query2: Are e-cigarettes safer than normal cigarettes
Query3: Can Harmone Replacement Therapy(HRT) cause cancer
Query4: Can MMR Vaccine lead to children developing autism
Query5: Should I take vitamin C for common cold

Table 2. 2016 FIRE CHIS dataset

	Query1	Query2	Query3	Query4	Query5
Dataset	341	413	246	259	278
Irrelevant	147	120	38	51	70
relevant	194	293	208	208	208

3.2 Baseline

We use Terrier 4.1[7] to perform the retrieval. TF-IDF model is chosen as the weighting model and parameters are set to default. Default Terrier term pipeline is used to remove the stop words and stem the words in queries as well as the training dataset. To get our baseline, we retrieve the dataset with the original queries from CHIS task.

[6] http://fire.irsi.res.in/fire/2016/home
[7] www.terrier.org

3.3 Query expansion

Expansion with PRF. We retrieve the dataset with the original query Q and get a ranked list of the documents which are regarded as relevant. We expand 10 terms T_{PRF} from the top 3 documents in the ranked list. The expanded query is noted as Q_{PRF}. We then retrieve the dataset with Q_{PRF}.

$$Q_{PRF} = \{Q, T_{PRF}\} \tag{1}$$

Expansion with UMLS. The dataset from CHIS is consumer focused and includes many common used words by laypeople. Considering the characteristic of the dataset, we extract synonyms and related medical terms both from MeSH and CHV vocabularies in our experiments. Terms T_{MeSH} expanded from MeSH and terms T_{CHV} from CHV are all included in to expand the original query. No terms selection algorithms are used in our experiments.

$$Q_{UMLS} = \{Q, T_{MeSH}, T_{CHV}\} \tag{2}$$

Expansion with Word2vec models. In our experiment, we train a Word2vec model with Wikipedia data. CBOW model is used for training the model. Original query is pre-processed by removing stop words and parsing the query into token terms. Pre-trained Word2vec model is used to find a list of words for every token term. We experiment on expanding every token term with the top 10 words, which shows the best performance in our experiments.

$$Q_{W2V} = \{Q, T_{W2V}\} \tag{3}$$

3.4 Results

Table 3,4,5 presents the results in the evaluation of precision, recall and F1 score accordingly, and Figure 4 depicts the comparison of the performance on different techniques. The results show that when compared with different evaluation measures, the performance can differ tremendously. Following we'll discuss our observations based on different evaluation measures.

Precision evaluation. From table 3 and the precision part of figure 4, we can see that for every query searching, when evaluated with precision, no other expansion techniques outperform baseline. We can conclude that working on this dataset, query expansion techniques will decline the precision in all cases compared to the baseline.

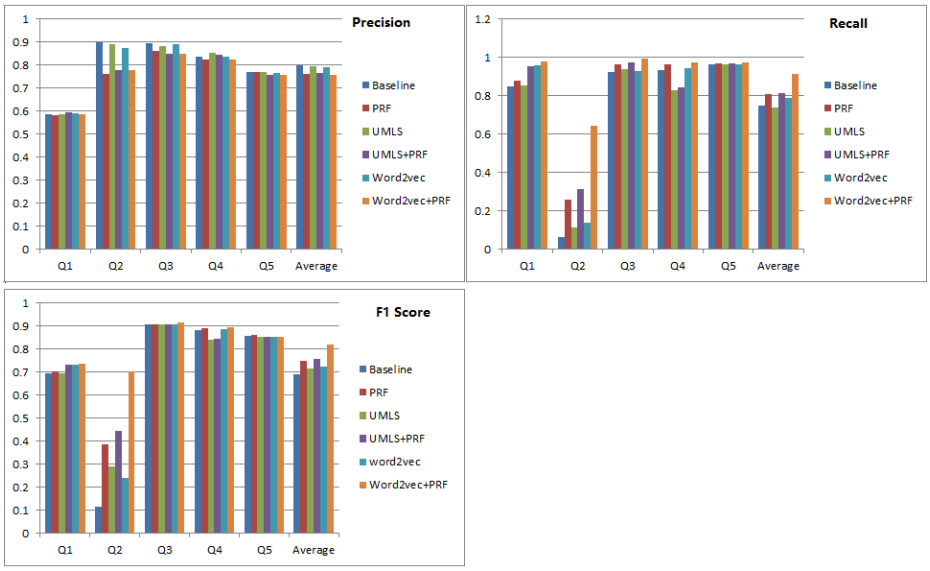


Fig. 4. Comparison of different query expansion techniques

Table 3. Precision on baseline and different query expansion techniques

	Baseline	PRF	UMLS	UMLS+PRF	Word2vec	Word2vec+PRF
Q1	0.587	0.584	0.585	0.593	0.592	0.588
Q2	0.9	0.76	0.892	0.778	0.872	0.777
Q3	0.893	0.862	0.882	0.849	0.889	0.848
Q4	0.836	0.826	0.852	0.846	0.838	0.825
Q5	0.77	0.769	0.769	0.759	0.764	0.758
Average	0.797	0.76	0.796	0.765	0.791	0.759

Recall evaluation. Table 4 and the recall part in figure 4 show that when evaluated with recall, expansion techniques show improvement on query 1,2,3,5 except query 4. Expansion with UMLS only or UMLS and PRF together decreases the performance on query 4. There is also a little bit decrease for using UMLs on query 5 and the average on all. From the results, we can see that PRF alone or PRF together with Word2vec can improve performance in all cases. UMLS alone or with PRF does not always improve the performance. Either UMLS or Word2vec, when using together with PRF, the results is better than working alone itself. Word2vec expansion technique works better than UMLS expansion when combined with PRF techniques.

Table 4. Recall on baseline and different query expansion techniques

	Baseline	PRF	UMLS	UMLS+PRF	Word2vec	Word2vec+PRF
Q1	0.851	0.881	0.856	0.954	0.959	0.979
Q2	0.061	0.259	0.113	0.311	0.14	0.642
Q3	0.923	0.962	0.938	0.976	0.928	0.995
Q4	0.933	0.962	0.827	0.846	0.942	0.976
Q5	0.966	0.97	0.962	0.971	0.966	0.976
Average	0.747	0.807	0.739	0.812	0.787	0.914

F1 score evaluation. The results in table 5 and F1 score in figure 4 show that PRF techniques improve the baseline in all 5 queries when evaluated with F1 score. The average improvement is about 5.82 % compared with the baseline. When using UMLS to expand the original query, the performance on query 2 and 3 get improved, but decrease on query 1, 4 and 5 compared with the baseline. When using UMLS expansion together with PRF techniques, we get an improved performance on query 1,2 and 3 but decreased one on query 4 and 5. We can also see that the average improvement of using a combined techniques is higher than using UMLS only. Moreover, we can see that a combined techniques of UMLS and PRF is better than using PRF individually in average. From the experiment results, we can see that UMLS does not improve the retrieval performance in all queries, although it improves the performance in average. Same as recall, we can also see that Word2vec expansion techniques works better than UMLS expansion when combined with PRF techniques. PRF techniques improve retrieval performance on all queries in our experiment. Also, a combined scheme of using PRF together with UMLS or Word2vec techniques exceeds using each of them alone. We can infer that PRF techniques can improve performance in all cases and works better when combined with other expansion techniques. Beside, from the results, we can see that expansion with UMLS does not always outperform the baseline. It improves performance in some queries and decrease in others. However, the average score of using UMLS expansion is still better than the baseline in our experiment. Using Word2vec expansion combined with

PRF techniques gets higher score than other techniques in our experiment. All expansion techniques have improved the performance in average compared with baseline.

Table 5. F1 score on baseline and different query expansion techniques

	Baseline	PRF	UMLS	UMLS+PRF	word2vec	Word2vec+PRF
Q1	0.695	0.702	0.695	0.731	0.732	0.735
Q2	0.115	0.387	0.29	0.444	0.241	0.703
Q3	0.908	0.909	0.909	0.908	0.908	0.916
Q4	0.882	0.889	0.839	0.846	0.887	0.894
Q5	0.857	0.86	0.854	0.852	0.854	0.853
Average	0.691	0.749	0.717	0.756	0.724	0.82

4 Discussion and Future Work

In our experiment, we have observed that not all the expanded terms are helpful and some even worsen the performance. Our observation also proves that thesaurus based query expansion technique does not always effective and the results can be mixed, which has been observed in early papers (Lu et al., 2009) (Darmoni et al., 2012) (Shen and Nie, 2015). It has been pointed out that expanding queries with Metathesaurus terms improved performance for certain instances (Hersh et al., 2000), yet the paper did not delineate those instances. Another paper (Voorhees and Hersh, 2012) stated that vocabulary normalization device must be used carefully as multiple groups also demonstrated that aggressive use harms baseline performance. A recent paper (Balaneshin-kordan et al., 2015) pointed out that only the concepts that belong to some specific kinds of semantic type can be included in the expanded query. Our future work will research into term selection schemes to select suitable terms for query expansion.

The performance of using Word2vec model to find close words as expansion exceeds other state of the art techniques in our experiment. We propose to continue the research into applying word2vec models and related techniques like neural network techniques into our work. Our future work in this area includes training a domain specific Word2vec models with domain medical data.

5 Acknowledgments

This work was supported by EACEA under the Erasmus Mundus Action 2, Strand 1 project LEADER - Links in Europe and Asia for engineering, eDucation, Enterprise and Research exchanges.

Bibliography

Aronson, A. R. and T. C. Rindflesch (1997). Query expansion using the umls metathesaurus. In *Proceedings of the AMIA Annual Fall Symposium*, pp. 485. American Medical Informatics Association.

Balaneshin-kordan, S., A. Kotov, and R. Xisto (2015). Wsu-ir at trec 2015 clinical decision support track: Joint weighting of explicit and latent medical query concepts from diverse sources. In *Proceedings of the 2015 Text Retrieval Conference*.

Carpineto, C. and G. Romano (2012). A survey of automatic query expansion in information retrieval. *ACM Computing Surveys (CSUR) 44*(1), 1.

Christopher, D. M. and S. Hinrich (1999). Foundations of statistical natural language processing. *MIT Press. Chunga, J. & Tan, FB (2004). Antecedents of computer playfulness: an exploratory study on user acceptance of general information-searching websites. Information & Management 41*(7), 869–881.

Darmoni, S. J., N. Griffon, and A. Névéol (2012). Improving information retrieval using medical subject headings concepts: a test case on rare and chronic diseases. *Journal of the Medical Library Association 100*(3), 176.

Hersh, W., S. Price, and L. Donohoe (2000). Assessing thesaurus-based query expansion using the umls metathesaurus. In *Proceedings of the AMIA Symposium*, pp. 344. American Medical Informatics Association.

Humphreys, B. L., D. A. Lindberg, H. M. Schoolman, and G. O. Barnett (1998). The unified medical language system. *Journal of the American Medical Informatics Association 5*(1), 1–11.

Lopes, C. T. and C. Ribeiro (2016). Effects of language and terminology on the usage of health query suggestions. In *International Conference of the Cross-Language Evaluation Forum for European Languages*, pp. 83–95. Springer.

Lu, Z., W. Kim, and W. J. Wilbur (2009). Evaluation of query expansion using mesh in pubmed. *Information retrieval 12*(1), 69–80.

Mikolov, T., K. Chen, G. Corrado, and J. Dean (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Palotti, J., G. Zuccon, L. Goeuriot, L. Kelly, A. Hanbury, G. Jones, M. Lupu, and P. Pecina (2015). Clef ehealth evaluation lab 2015, task 2: Retrieving information about medical symptoms. In *Proc. of CLEF*.

Shen, W. and J.-Y. Nie (2015). Is concept mapping useful for biomedical information retrieval? In *International Conference of the Cross-Language Evaluation Forum for European Languages*, pp. 281–286. Springer.

Song, Y., Y. He, Q. Hu, L. He, and E. M. Haacke (2015). Ecnu at 2015 ehealth task 2: User-centred health information retrieval. *Proceedings of the ShARe/CLEF eHealth Evaluation Lab*.

Srinivasan, P. (1996). Query expansion and medline. *Information Processing & Management 32*(4), 431–443.

Voorhees, E. M., D. K. Harman, et al. (2005). *TREC: Experiment and evaluation in information retrieval*, Volume 1. MIT press Cambridge.

Voorhees, E. M. and W. R. Hersh (2012). Overview of the trec 2012 medical records track. In *TREC*.

Wang, C., L. Cao, and B. Zhou (2015). Medical synonym extraction with concept space models. *arXiv preprint arXiv:1506.00528*.

Zhu, D. and B. Carterette (2012). Exploring evidence aggregation methods and external expansion sources for medical record search. Technical report, DTIC Document.

A multi-classifier approach versus a single classifier approach in the identification of modality

João Sequeira
d11594@alunos.uevora.pt

Department of Informatics, University of Évora, Portugal

Abstract This paper is a continuation in the study of automatic modality tagging for Portuguese language. Our main goal is to study the use of single classifiers, where we have eleven modal classes from eleven modal verbs, with and without lemma information, compare it with a classification made by eleven different classifiers (multi-classifier), each one with the modality values present in each verb corpus. The performance was measured using precision, recall and F_1. Due to the difference between number of examples for each class we used weighted average approach of each performance metric. The system used a bow approach as baseline and compared it with the set path + context attributes, using different kernels, rbfkernel and polykernel with exponents from 1 to 3. In some cases the bow had similar or better values, depending on the kernel used. The best F_1 values are above .60, the difference between a single classifier approach and one with a set of individual classifiers (multi-classifier) can be considered insignificant.

1 Introduction

In the last years there was a great evolution in fields related to machine learning in the pursuit of forms of artificial intelligence (AI), becoming this a major trend of both academic and companies research, like Google and Microsoft. The fields are diverse, financial fraud identification, image recognition and even AI's that can rewrite their own code or write other programs [2,3,4]. One of these fields is the understanding of texts, information extraction and classification, called Natural Language Processing (NLP). Currently, inside NLP one of the most important sub-field to proliferate focuses on sentiment analysis and opinion mining. One step of it, is the search of an automatic way to distinguish between the factual and non-factual nature of events and the detection of subjective perspective underlying texts. Modality is one indicator of subjectivity and factuality in texts, it is usually defined as the expression of the speaker's opinion and of his attitude towards the proposition [14]. Traditionally, covers epistemic modality, which is related to the degree of commitment of the speaker to the truth of the proposition (whether the event is perceived as possible, probable or certain), but also deontic modality (obligation or permission), capacity and volition [19].

This paper was made in the context of PhD Seminar II.

This paper, followed the experiments done by Sequeira *et al.* [19], in the pursuit of a way to create a semi-automatic tagging system of modality for a larger corpus of Portuguese language, using a small manual annotated corpus with a modality scheme for Portuguese [9].

Like the previous experiments, due to their polysemy (expression of more than one type of modality), the eleven modal verbs (also called, trigger) of Portuguese studied in this paper are: *arriscar* (chance/risk/ dare), *aspirar* (aspire), *conseguir* (manage to/succeed in/be able to), *considerar* (consider/regard), *dever* (shall/might), *esperar* (wait/expect), *necessitar* (need/require), *permitir* (allow/permit), *poder* (may/can), *precisar* (need) and *saber* (know).

To explain polysemy, we can use the verb *poder*, that can have:

- `Epistemic`, stating that something is possible;
- `Deontic`, denoting a permission, or;
- may express an `Internal Capacity`, the fact that someone is able to do something.

Our experiment will use the set of attributes described in [19]: target, path and context. From all the combinations between sets it will be compared the best one (path + context) with a bag of words approach. The results presented were obtained with the following variations of the corpus:

- *Exp A*: one classifier with the examples of eleven verbs with all types of modality, also eleven. It was added one extra attribute that was the lemma of the trigger;
- *Exp B*: one classifier with the examples of eleven verbs with all types of modality, also eleven. Without the lemma of the trigger;
- *Exp C*: eleven individual classifiers, one for each verb, trained with the modality types associated with each verb in the corpus.

The experiments were run using a Support Vector Machine (SVM) [21] algorithm called Sequential Minimal Optimization (SMO) [15]. The results are expressed with precision, recall and F_1 measure.

The paper is structured as follows: Section 2 presents related work on the automatic annotation of modality, Section 3 presents the developed system with the experimental setup, corpus information, attributes used and performance calculation, Section 4 presents the experiments done and the results achieved. Finally, in Section 5 we withdraw some conclusions and present some future work that would improve our system.

2 Related work

Most of the work done related to modality are for English language. This means that it's a new field in the Portuguese Language, although Portuguese is one of the 10 most spoken languages in the world, with more than 260 millions of speakers [1].

In the works done by Baker *et al.* [5], Matsuyoshi *et al.* [12], Nirenburg and McShane [13] and Sauri*et al.* [18] can be seen annotation schemes for modality applied to English language. For Portuguese we can identify works from Hendrickx *et al.* [9] for written European Portuguese and Ávila *et al.* [22] for spoken Brazilian Portuguese.

Baker *et al.* [5] tested two rule-based modality taggers that identify both the modal trigger (word or word list where modality is expressed, usually by the use of modal verbs) and its target (is the event, state, or relation, over which the modality has scope) and achieve results of 86% precision for a standard LDC data set.

Thompson *et al.*. [20] addressed the identification of expressions linked to modality in biomedical texts using three dimensions: the kind of knowledge, level of certainty and point of view. Their approach is characterized by using a list of words and phrases with modal characteristics specific for the biomedical domain.

Ruppenhofer and Rehbein [17] developed a modal verb annotation scheme for news articles written in English. The system used a classifier of maximum entropy [16] to identify the verbs *can*, *may/might*, *must* and *should*. The attributes used were divided into three categories: (i) target/verb; (ii) context; (iii) path. They used different combinations of attributes with different context sizes and the results were compared to those of a baseline system always assigning the most common value to each verb. The best result was achieved for the verb *must* with an accuracy of value 93,50%, followed by the verb *shall/should* with 91,61%, *may/might* with 85,71% and finally *can* with 68,70%.

Sequeira *et al.*. [19] have presented a study of the same eleven verbs, and can be seen as a early study to the work presented in this paper. The goal was to create a set of attributes for the creation of automatic taggers and compare them with a the bag-of-words (bow) approach. The paper covers the creation of the corpus (composed by eleven verbs), the use of a parser to extract syntactic and semantic information from the sentences and a machine learning approach to identify modality values.

3 Developed system

The developed system is based in the one presented in Sequeira *et al.* [19]. The pre-processing of data was made by tagging modality by human annotation and the syntactic analysis was performed using the PALAVRAS parser [6]. It was used a set of sentences that include modal verbs to build the data for the Machine Learning algorithm. The set of attributes are the same and based in the results obtained in Sequeira *et al.* [19], the tests in this work were made only with the set *path + context* and bow as baseline.

3.1 Corpus and dataset

The corpus used is composed by 936 sentences from eleven verbs and have eleven different types of modality values.

Table 1. Corpus characterization: class and number of sentences per modal value.

class	modal type	number of sentences
0	effort	20
1	epistemic belief	66
2	volition	60
3	deontic permission	68
4	deontic obligation	97
5	participant-internal necessity	87
6	evaluation	11
7	participant-internal capacity	94
8	epistemic possibility	299
9	success	41
10	epistemic knowledge	93
	total	936

In Table 1 is represented the class and number os examples associated with each modal value. The number os sentences are discrepant between modal values, for example *epistemic possibility* has 299 examples while *evaluation* has 11, or we can see that most classes have a number of examples above 40 but *effort* and *evaluation* have less than half of it. Even if only the main classes were considered, *epistemic*, *deontic* and *volition* the values would be discrepant between them.

In Table 2 is represented the number of examples for each modal value presented in each verb. There are verbs with two modal values like *aspirar*, *considerar*, *permitir*, *precisar* and *saber*, three modal values like *arriscar*, *conseguir*, *esperar* and *necessitar*, and four modal values like *dever* and *poder*. In almost all verbs the number of examples for each modal value are discrepant, in some cases they might not even be considered. Inside the verbs are also discrepant values between the classes, using as example the verb *saber*, it has two classes, one with 93 examples and another with 10, or *necessitar* that have one class with 41 examples and other two with less than 10.

3.2 Feature extraction

The attributes sets created were the same used by Sequeira *et al.* [19] based on the work done by Ruppenhofer and Rehbein [17]. Following their approach, we had three sets of attributes: *(i) trigger*, related to the modal verb information; *(ii)context*, related to context surrounding the trigger (based on the previous experiments to this work, it was considered a window of size five, with the trigger in central position); *(iii) path*, related to the information extracted from the path done from the trigger to the root on the syntactic and morphological analysis tree.

For the creation of the syntactic and morphological analysis trees of the examples was used the PALAVRAS parser [6,7]. The syntactic and morphological analysis trees were the source of information to build the attributes for each set.

Table 2. Corpus characterization: number of sentences per modal value for each verb.

verb	modal classes	modality type	number of sentences	total
arriscar	0	effort	20	46
	1	epistemic belief	1	
	8	epistemic possibility	25	
aspirar	1	epistemic belief	18	52
	2	volition	34	
conseguir	7	participant-internal capacity	42	87
	8	epistemic possibility	4	
	9	success	41	
considerar	1	epistemic belief	15	26
	6	evaluation	11	
dever	1	epistemic belief	2	124
	3	deontic permission	3	
	4	deontic obligation	78	
	8	epistemic possibility	41	
esperar	1	epistemic belief	30	57
	2	volition	26	
	8	epistemic possibility	1	
necessitar	4	deontic obligation	8	51
	5	participant-internal necessity	41	
	7	participant-internal capacity	2	
permitir	3	deontic permission	19	80
	8	epistemic possibility	61	
poder	3	deontic permission	46	254
	4	deontic obligation	1	
	7	participant-internal capacity	40	
	8	epistemic possibility	167	
precisar	4	deontic obligation	10	56
	5	participant-internal necessity	46	
saber	7	participant-internal capacity	10	103
	10	epistemic knowledge	93	

Table 3 summarizes the attributes extracted. For the *trigger* set, it was considered the information from the trigger itself and from the ancestors. For the *path* set, it was considered the information about the siblings of the trigger and the path from the trigger to root. For the *context* set, it was considered information about the words to the left and right of the trigger.

Table 3. Attributes extracted from trigger, path and context. (Source: [19])

trigger		**path**		**context**	
source	attributes	source	attributes	source	attributes
trigger	POS function role morphological semantic	siblings	POS function role morphological semantic	left/right trigger	POS word lemma
ancestors	POS function	trigger to root	POS function		

3.3 Experimental setup

Linear classifiers separate the data using hyperplanes; in presence of a binary problem with x_i and y_i being the attribute vector and the class of example i [10], we have

$$(x_i, y_i), \ldots, (x_n, y_n),\ x_i \in \mathbb{R}^n \qquad y_i \in \{+1, -1\} \tag{1}$$

and the separating hyperplane is

$$(w \cdot x) + b = 0, \quad w \in \mathbb{R}^n, \quad b \in \mathbb{R} \tag{2}$$

Support Vector Machines are linear classifiers that try to create an optimal separating hyperplane with the biggest margin (region that separates positive from negative examples). The properties that make this algorithm so attractive in classification tasks are a well-defined theoretical basis, robustness and good generalization ability [21,11].

Based on previous tests [19] the chosen algorithm was the SMO [15].

Different sets of extracted attributes were evaluated and compared with a typical bag-of-words approach. For the evaluation we used a 5-fold stratified cross-validation procedure and computed average precision, recall and F1 performance measures. Appropriate statistical tests with 95% of significance were applied to analyse the differences between results.

It were compared different types of experiments, one single classifier (eleven verbs) with the set *path + context* with the lemma of the trigger, one single classifier (eleven verbs) with the set *path + context* without the lemma information and a multi-classifier approach with eleven individual classifiers (one for each verb) with the set *path + context*. It was used a bow approach as a baseline for comparison

These machine learning experiments were conducted using Weka framework [8].

3.4 Performance calculation

The system performance was calculated using, precision, recall and F_1. As it was showed in Table 1 and Table 2, the number of classes examples in global and inside each verb have very discrepant values. Because of it, the calculation of precision, recall and F_1 have to take this into account. Instead of having a arithmetic mean of the values of precision, recall and F_1 as final values, it was calculated a weighted average of each one.

For *Exp A* and *Exp B*, Weka made the performance calculation by itself. For *Exp C*, the final values were calculated by hand following a similar approach of Weka, precision using equation 3, recall using equation 4 and F_1 score using equation 5. Being x_i the number of examples a class have $(x_0, \ldots, x_{10})$, P_i the precision obtained with each class $(P_0, \ldots, P_{10})$, the recall obtained with each class $(R_0, \ldots, R_{10})$ and F_1 score obtained with each class $(F0_1, \ldots, F10_1)$.

$$precision = \frac{(x_0 \times P_0) + (x_1 \times P_1) + \cdots + (x_{10} \times P_{10})}{x_0 + x_1 + \cdots + x_{10}} \tag{3}$$

$$recall = \frac{(x_0 \times R_0) + (x_1 \times R_1) + \cdots + (x_{10} \times R_{10})}{x_0 + x_1 + \cdots + x_{10}} \tag{4}$$

$$F_1 = \frac{(x_0 \times F0_0) + (x_1 \times F1_1) + \cdots + (x_{10} \times F10_{10})}{x_0 + x_1 + \cdots + x_{10}} \tag{5}$$

4 Experiments

In order to evaluate the performance of each approach, we made variations in the configurations of the SMO algorithm, also in the attributes and type of classifier. The variations made are expressed bellow:

- sets:
  - *path + context*;
  - *bow*;
- kernel:
  - *polyKernel* with exponent (1, 2 and 3);
  - *rbfKernel*;
- experiments:
  - *A*: one single classifier with lemma of the trigger;
  - *B*: one single classifier without lemma of the trigger;
  - *C*: eleven classifiers in a multi-classifier approach, one for each verb.

Tables 4, 5 and 6 present the precision, recall and F_1 values, respectively.

4.1 Discussion of results

For precision, Table 4, in the *Exp A* the range are between .544 with *rbfKernel* and .659 *polyKernel (e = 1)*, with the set *path + context* and between .32 using *polyKernel (e = 2)* and .691 with *polyKernel (e = 1)*. In the *Exp B* the range are between .285 with *rbfKernel* and .447 *polyKernel (e = 2) and (e = 3)*, with the set *path + context* and between .102 using *rbfKernel* and .355 with *polyKernel (e = 1)*. In the *Exp C* the range are between .615 with *rbfKernel* and .689 *polyKernel (e = 2)*, with the set *path + context* and between .605 using *rbfKernel* and .681 with *polyKernel (e = 1)*.

Almost in all experiments, the *rbfKernel* had one of the worse performances, and *polyKernel* with exponent 2 or 3 had the best ones.

Using the lemma attribute have influence on te precision result, using *path + context*, when compared the *Exp A* with *Exp B*, the values are higher, an average .2 better, but when compared with *Exp C* the values don't have a meaningful difference, .03 bellow. Using *bow*, when compared the *Exp A* with *Exp B*, values are also higher, more than double, when compared with *Exp C* the values are almost equal but in the best case is .01 better.

Table 4. Precision for each variation of kernel and type of classifier for the bow and path + context approach.

attribute	type of kernel	Exp A	Exp B	Exp C
path + context	polyKernel (e = 1)	.659	.42	.678
	polyKernel (e = 2)	.627	.447	.689
	polyKernel (e = 3)	.582	.447	.678
	rbfKernel	.544	.285	.615
bow	polyKernel (e = 1)	.691	.355	.681
	polyKernel (e = 2)	.533	.261	.652
	polyKernel (e = 3)	.32	.266	.612
	rbfKernel	.424	.102	.605

For recall, Table 5, in *Exp A* the *bow* approach obtained a value .03 better than *path + context*. In *Exp B* and *Exp C* the values of *path + context* were better than using *bow*.

The values obtained with *Exp A* are better than the ones of *Exp B*, this indicates that using information of lemma is essential to improve results, in some cases above .2 for the same algorithm (*polyKernel (e = 1)*), and using the same algorithm with *bow* the difference are bigger, above .3.

Exp C got best recall values, .698 for *path + context* with *polyKernel (e = 2)* and .689 for *bow* with *polyKernel (e = 1)*.

Although without many significant discrepancy the recall values from *Exp C* are better than the ones presented using *Exp A*. Even with *rbfKernel*, that obtained consistently the lowest values, *Exp C* got a value similar to the best one of *Exp A*. This show that, without many difference between them, using eleven

different classifiers, the system obtained better recall values than using a single classifier with information related to all verbs.

Table 5. Recall for each variation of kernel and type of classifier for the bow and path + context approach.

attribute	type of kernel	Exp A	Exp B	Exp C
path + context	polyKernel (e = 1)	.675	.436	.678
	polyKernel (e = 2)	.652	.475	.698
	polyKernel (e = 3)	.584	.453	.693
	rbfKernel	.53	.408	.673
bow	polyKernel (e = 1)	.708	.385	.689
	polyKernel (e = 2)	.578	.314	.667
	polyKernel (e = 3)	.353	.121	.640
	rbfKernel	.465	.319	.668

For F_1, Table 6, in *Exp A* the *bow* approach obtained a value .019 better than *path + context*, both using *polyKernel (e = 1)*. In the *Exp B* and *Exp C* the values of *path + context* were better than with *bow*.

F_1 values being a relation between precision and recall, it was normal to expect better values for *Exp A* when compared with *Exp B* proving that when lemma information was added to the initial attributes, the performance of the classifier was improved. F_1 values improved in some cases almost .25 when we use *polyKernel (e = 1)* as comparison

Of all kernel tests, *polykernel* and *rbfkernel*, *Exp C* in general got the best values of F_1, both with *bow* and *path + context*, all above .61. *Exp A* got similar values when compared with *Exp C*, but got a clearly poor performance with both sets using *polyKernel (e = 3)* and *rbfkernel*.

Table 6. F_1 for each variation of kernel and type of classifier for the bow and path + context approach.

attribute	type of kernel	Exp A	Exp B	Exp C
path + context	polyKernel (e = 1)	.664	.426	.673
	polyKernel (e = 2)	.62	.433	.678
	polyKernel (e = 3)	.536	.386	.664
	rbfKernel	.425	.279	.628
bow	polyKernel (e = 1)	.683	.327	.658
	polyKernel (e = 2)	.536	.223	.627
	polyKernel (e = 3)	.275	.129	.616
	rbfKernel	.333	.155	.627

5 Conclusions and future work

With this work, we tried to keep addressing the topic of researching automatic approaches to tagging modality in Portuguese language. Being a field linked to NLP topics as sentiment analysis and opinion mining, it is important to be done with the best performance possible.

Following works done before, we used eleven modal verbs with morphological, syntactic and some semantic attributes from the sentences, obtained with PALAVRAS.

Using Weka framework, we conducted several experiments, with two sets of attributes, *bow* as a baseline and *path + context*. We used different types of kernel *rbfkernel* and *polykernel* with a variation of exponents. It was also tested three approaches, one single classifier for all verbs with lemma information, one single classifier for all verbs without lemma information and another with eleven different classifiers (multi-classifier approach).

Comparing the performance of our system with each set of algorithm and attributes, we can conclude that adding lemma information improves the performance. Comparing the single classifier using lemma with the multi-classifier the results don't have significant difference but the last have better results. With the individual classifiers all values were above .60, with *bow* and *path + context*, very different from the single classifier values where *rbfkernel* got lower results and *polykernel* with exponent one normally got the better ones.

The corpus is small and with unbalanced number of examples for each class, this can also be an influence in the performance of the system.

This is a further step in the creation of a semi-automatic modality tagging system, so as future work, we are about to start addressing the identification of the subject linked to modality expressed by the trigger verb. In order to improve the performance or to exclude the number of examples between classes as potential limitation for future studies it is necessary to balance the number of classes examples, increasing the size of the corpus.

References

1. As 10 línguas mais faladas no mundo, `https://observalinguaportuguesa.org/as-linguas-mais-faladas-no-mundo/`
2. Google's ai is learning to make other ai, `https://futurism.com/googles-ai-is-learning-to-make-other-ai`
3. Microsoft's ai is learning to write code by itself, not steal it, `https://qz.com/920468/artificial-intelligence-created-by-microsoft-and-university-of-cambridge-is-learning-to-write-code-by-itself-not-steal-it`
4. New ai can write and rewrite its own code to increase its intelligence, `https://futurism.com/new-ai-can-write-and-rewrite-its-own-code-to-increase-its-intelligence/`
5. Baker, K., Bloodgood, M., Dorr, B., Filardo, N.W., Levin, L., Piatko, C.: A modality lexicon and its use in automatic tagging. In: Chair), N.C.C., Choukri, K., Maegaard, B., Mariani, J., Odijk, J., Piperidis, S., Rosner, M., Tapias, D. (eds.)

Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10). European Language Resources Association (ELRA), Valletta, Malta (may 2010)
6. Bick, E.: The parsing system PALAVRAS. Aarhus University Press (1999)
7. Bick, E.: The Parsing System "Palavras". Automatic Grammatical Analysis of Portuguese in a Constraint Grammar Framework. University of Arhus, Århus (2000)
8. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The weka data mining software: An update. SIGKDD Explor. Newsl. 11(1), 10–18 (Nov 2009)
9. Hendrickx, I., Mendes, A., Mencarelli, S.: Modality in text: a proposal for corpus annotation. In: Chair), N.C.C., Choukri, K., Declerck, T., Doğan, M.U., Maegaard, B., Mariani, J., Moreno, A., Odijk, J., Piperidis, S. (eds.) Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12). European Language Resources Association (ELRA), Istanbul, Turkey (may 2012)
10. Kudoh, T., Matsumoto, Y.: Use of support vector learning for chunk identification. In: CoNLL'00 – 4th Conference on Computational Natural Language Learning. pp. 142–144 (2000)
11. Lorena, A., Carvalho, A.: Introdução às máquinas de vectores suporte (support vector machines). Relatórios técnicos do ICMC (Abril 2003)
12. Matsuyoshi, S., Eguchi, M., Sao, C., Murakami, K., Inui, K., Matsumoto, Y.: Annotating event mentions in text with modality, focus, and source information. In: Chair), N.C.C., Choukri, K., Maegaard, B., Mariani, J., Odijk, J., Piperidis, S., Rosner, M., Tapias, D. (eds.) Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10). European Language Resources Association (ELRA), Valletta, Malta (may 2010)
13. Nirenburg, S., McShane, M.: Annotating modality. Tech. rep., University of Maryland, Baltimore County, USA (March 2008)
14. Palmer, F.R.: Mood and Modality. Cambridge textbooks in linguistics, Cambridge University Press (1986)
15. Platt, J.C.: Advances in kernel methods. chap. Fast Training of Support Vector Machines Using Sequential Minimal Optimization, pp. 185–208. MIT Press, Cambridge, MA, USA (1999)
16. Ratnaparkhi, A.: A maximum entropy model part-of-speech tagger. In: EMNLP'96 – Empirical Methods in Natural Language Processing Conference. pp. 133–141 (1996)
17. Ruppenhofer, J., Rehbein, I.: Yes we can!? annotating english modal verbs. In: Chair), N.C.C., Choukri, K., Declerck, T., Doğan, M.U., Maegaard, B., Mariani, J., Odijk, J., Piperidis, S. (eds.) Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12). European Language Resources Association (ELRA), Istanbul, Turkey (may 2012)
18. Sauri, R., Verhagen, M., Pustejovsky, J.: Annotating and recognizing event modality in text. In: FLAIRS Conference. pp. 333–339 (2006)
19. Sequeira, J., Gonçalves, T., Quaresma, P.: Using syntactic and semantic features for classifying modal values in the portuguese language. In: CICLing 2016, 17th International Conference on Intelligent Text Processing and Computational Linguistics
20. Thompson, P., Venturi, G., McNaught, J., Montemagni, S., Ananiadou, S.: Categorising modality in biomedical texts. In: Proceedings of the LREC 2008 Workshop on Building and Evaluating Resources for Biomedical Text Mining. pp. 27–34. Marrakech, Marrocos (2008)

21. Vapnik, V.N.: Statistical Learning Theory. Wiley-Interscience (1998)
22. Ávila, L., Melo, H.: Challenges in modality annotation in a brazilian portuguese spontaneous speech corpus. In: Proceedings of IWCS 2013 WAMM Workshop on the Annotation of Modal Meaning in Natural Language. Association for Computational Linguistics, Postam, Germany (2013)

Age and Gender Classification of Tweets using Convolutional Neural Networks

Roy Bayot

Universidade de Évora, Department of Informatics,
Rua Romão Ramalho n°59, 7000-671 Évora, Portugal
d11668@alunos.uevora.pt

Abstract. This paper describes experiments that use convolutional neural networks together with word2vec word embeddings. The network consists of five layers and is trained using adadelta. It starts with an embedding layer where a word is represented by a vector, followed by a convolutional layer composed of three filters, each with 100 feature maps. It is followed by a max-over-time pooling layer which is done on each map and the resulting features are concatenated before a dropout layer and a softmax layer. The network was trained to classify age and gender for English and Spanish using tweets and it gave results that outperform previous experiments. The highest English age and gender classification accuracy obtained are 49.6% and 72.1% respectively. The highest Spanish age and gender classification accuracy obtained on the other hand are 56.0% and 69.3% respectively.

Keywords: author profiling, twitter, word vectors, word2vec, convolutional neural networks

1 Introduction

Social media has grown rapidly in the recent years, especially with the advent of sites like Facebook, Instagram, Twitter, and Snapchat. New communication models have arisen from the surge of social media. Logging worker tasks and productivity for instance could be done through Yammer. Another example would be team communication using Slack or Telegram. However, even with the proliferation of these different communication models and relations, there can still be a problem of incomplete information with regards to the person writing the content. Determining a person's traits based on the text they wrote is known as author profiling and it has been of increasing importance in the recent years. This could be seen in areas such as digital forensics where linguistic profiles could be used to check for criminals or in business intelligence for targeted advertising or product reviews and analysis.

This paper would be submitted for Seminars 3.

The work tries to solve author profiling for age and gender in English and Spanish with twitter text by using convolutional neural networks. It follows the work of Kim [9] with some minor modifications. Aside from using convolutional neural networks with word vectors, the work also aims to observe the effect of the size of the vector dimensions to the accuracy. It also aims to observe if tuning pre-trained vectors would improve the result. Finally, the work compares the current accuracy results to the results from previous experiments.

The paper is organised as follows. Section 2 covers related literature where it initially discussed previous author profiling endeavors, then followed by methods in PAN, followed by an explanation on word2vec, uses of convolutional neural networks, and finally a short summary on the previous experiment to which this work was compared. Section 3 describes the methodology, from the dataset, to the creation of the word2vec vectors, to details of the convolutional neural network architecture, the different variations, and then how it was evaluated. Section 4 gives the results and discussion while section 5 gives the conclusion and recommendations.

2 Related Literature

In previous author profiling research, most of the work is centered on hand crafted features as well as content-based and style-based ones. For instance, in the work of Argamon et al. in [2] where texts were categorized based on gender, age, native language, and personality, different content-based features and style-based features were used. Content-based features used were the 1000 words that appear frequently in the corpus which has the highest information gain to differentiate between classes. The style-based features included the nodes of a taxonomic tree made from systemic functional linguistics [7] with each value giving the frequency of the node's occurence normalized by the number of words in the text. Another example is the work of Schler et al. in [27] where writing styles in blogs are related to age and gender. Stylistic and content features were extracted from 71,000 different blogs and a Multi-Class Real Winnow was used to learn the models to classify the blogs. Stylistic features included parts-of-speech tags, function words, hyperlinks, and non-dictionary words; content features included word unigrams with high information gain.

2.1 PAN Editions

One particular initiative dealing with author profiling is PAN. In the first edition of PAN [21] in 2013, the task was age and gender profiling for English and Spanish blogs. In PAN 2014 [21], the task was profiling authors with text from four different sources - social media, twitter, blogs, and hotel reviews. In PAN 2015 [22], the task was limited to tweets but expanded to different languages with age and gender classification and a personality dimension. The languages include English, Spanish, Italian, and Dutch. There were 5 different personality

dimensions - extroversion, stability, agreeableness, conscientiousness, and openness. The more recent edition, PAN 2016 [23] deals with cross-genre evaluation where classifiers are trained on English, Spanish, and Dutch tweets while tested on other genres such as blogs, reviews, and other forms of social media.

Most of the methods include extracting content-based features such as bag of words, named entities, dictionary words, slang words, contractions, sentiment words, and emotion words; another would be stylistic features such as frequencies, punctuations, POS, HTML use, readability measures, and other various statistics. There are also features that are n-grams based, IR-based, and collocations-based; named entities, sentiment words, emotion words, and slang, contractions and words with character flooding were also considered. Some variations would be that of Maharjan et al. [14], where n-grams were used with stopwords, punctuations, and emoticons, and idf count was also used before placed into a classifier. In [31], different features were used that were related to length (number of characters, words, sentences), information retrieval (cosine similarity, okapi BM25), and readability (Flesch-Kincaid readability, correctness, style). Another approach was to use term vector model representation as in [30]. On the other hand, Marquardt et al. in [15], used a combination of content-based features (MRC, LIWC, sentiments) and stylistic features (readability, html tags, spelling and grammatical error, emoticons, total number of posts, number of capitalized letters number of capitalized words). Classifiers also varied for this edition; there was the use of logistic regression, multinomial Naïve Bayes, liblinear, random forests, Support Vector Machines, and decision tables.

In most of the editions, the work of Lopez-Monroy et al. in [12] provided a framework that works best for most tasks in most editions. They placed second for both English and Spanish in 2013 where they used second order representation based on relationships between documents and profiles. The work of Meina et al. [16] used collocations and placed first for English while the work of Santosh et al. in [26] worked well with Spanish using POS features in the same year. In the following year, the work of Lopez-Monroy et al. in [13] which uses the same method as the previous year [12] gave the best result with an average accuracy of 28.95% on all corpus-types and languages.

In 2015, the work of Alvarez-Carmona et al. [1] gave the best results on English, Spanish, and Dutch; their work used second order profiles as in the previous years as well as LSA. On the other hand, the work of Gonzales-Gallardo et al. [6] gave the best result for Italian; this used stylistic features represented by character n-grams and POS n-grams.

2.2 Word2vec

The overall theme is that hand-crafted features are extracted from the text and used into a classifier to predict. However there is a more recent trend where the system learns suitable filters at run time and uses the learned filters to generate a feature representation suitable for classification. This approach usually begins with learning word embeddings because it captures semantic information

between words that could be later leveraged. One of the more prominent embeddings is word2vec by Mikolov in [17] [18]. Essentially, words from a dictionary of a given corpus are initially represented with a vector of random numbers. A word's vector representation is learned by predicting it through its adjacent words; the basis for the words order is in a large corpus. Obtaining the vector can be done in two different ways - skip grams and continuous bag of words (CBOW). In CBOW, the word vector is predicted given the context of adjacent words; in skip grams, the context words are predicted given a word. The word vectors are then updated after all the predictions and will result in vectors that are not just random but have some semantic relation to each other. For instance, given a vector of king, man, and woman, doing vector operations such as $v_{king} - v_{man} + v_{woman}$ yields a vector closest to v_{queen}.

2.3 SKIMA

We used word embeddings in age and gender classification on the same dataset in our previous paper [3]. We experimented with using word2vec and support vector machines, where one training example is the average of the word vectors taken from all the tweets made by one user. We compared accuracy results between *tfidf* against 100 dimension word2vec vectors trained on continuous bag of words. Word2vec performs better than the usual *tfidf* for the given task. Additional experiments also showed that vectors which used the skip-grams method performed better than that which used continuous bag of words.

2.4 Convolutional Neural Networks

Averaging the vectors still seemed somewhat crude since all the vectors are given the same weight and filters are not learned to see which feature is necessary. We then look into neural network architectures that uses word vectors to find suitable filters for classification tasks. One such architecture is that of LeCun in [10]. The original paper works on images however it was adapted to work on text. For instance, the work of Kalchbrenner et al. uses convolutional neural networks to model sentences by using a dynamic k-max pooling [8]. On the other hand, convolutional neural networks were used for semantic parsing in the work of Yih et al. in [33] while Shen et al. used it for search query retrieval in [28]. The work done in this paper however closely follows that of Kim [9]. He used word vectors together with convolutional neural network on multiple benchmarks: movie reviews with one sentence per review in [20], Stanford Sentiment Treebank (neutral reviews removed and only binary labels), subjectivity dataset in [19], TREC question dataset in [11], customer reviews in [11], and opinion polarity detection of the MPQA dataset in [32]. This approach was able to improve the state of the art on five out of seven benchmarks - everything except TREC question dataset and the subjectivity datasets. The details of the network he used is described in section 3.4.

3 Methodology

The figure 1 given below shows an overview description of the system from how the dataset is manipulated before fed into the convolutional neural network and how it is evaluated. The details are described in the following subsections.

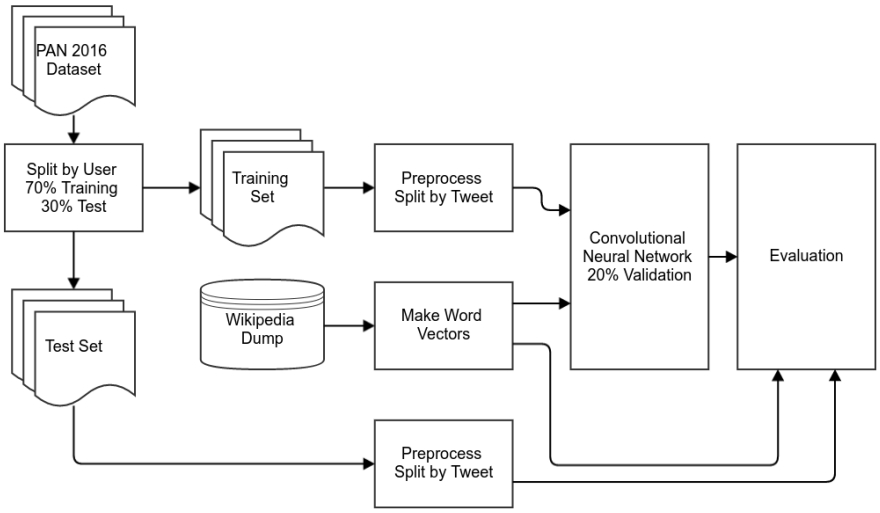


Fig. 1. Overview of the system.

3.1 Dataset

The dataset comes from PAN 2016 Author Profiling task [24]. It is composed of tweets from English, Spanish, and Dutch with profiling elements of age and gender. The categories for age classification are 18-24, 25-34, 35-49, 50-64, and 65 and above. Dutch does not have age information so we will not be using it. The tables 1 and 2 show information about the dataset. The dataset was then split with 70% to be used for training while the remaining 30% was held out for testing.

3.2 Preprocessing

All XMLs files from each user are read. This is done for both the training and test set. Then the tweets taken from each user are extracted to form one training example. The examples are then transformed by putting them all in lower case.

Table 1. Age and gender distribution for PAN 2016 dataset.

		English	Spanish
gender			
	male	218	125
	female	218	125
age			
	18-24	28	16
	25-34	139	64
	35-49	182	126
	50-64	80	38
	65-xx	6	6

Table 2. Basic Statistics for PAN 2016 Dataset

	English		Spanish	
	Train	Test	Train	Test
Total Number of Users	306	130	175	75
Total Number of Tweets	194953	82839	151362	57258
Total Number of Nans Removed	1963	1371	1986	55
Max Number of Tokens in Tweet	69	70	98	47
Min Number of Tokens in Tweet	1	1	1	1
Average Tweet Length	13.15	13.15	13.84	13.92
Standard Deviation	6.21	6.32	6.25	6.19
Mode	11	10	18	16

No stop words are removed. Hash tags, numbers, mentions, shares, and retweets were not processed or transformed to anything else. They were retained as is and therefore will correspond to another item in the dictionary of features. The test set will be set aside for the final evaluation while the training set will be used to train the network.

3.3 Pre-trained Vectors

Before training begins, word embeddings need to be created. The wikipedia dump from February 05, 2016 was used. The English wikipedia dump at that time was 11.8Gb compressed with bzip. The Spanish dump on the other hand had 2.2Gb compressed. This dump was then extracted and transformed such that everything was turned into lowercase and entries are in one file. This was then used as input to the word2vec implementation of gensim [25] to generate our own vectors. Regarding word2vec parameters, no lemmatization was done, and 5 was the window size used. We also used skip grams instead of continuous bag of words and finally, the output dimensions were 100 and 300.

3.4 Model

The model architecture is shown in figure 2. This is similar to the architecture of Kim [9] which is a variant of the architecture given by Collobert et. al. [5] and is implemented in Keras [4] with a Theano [29] backend ran on an NVIDIA Tesla K20c GPU.

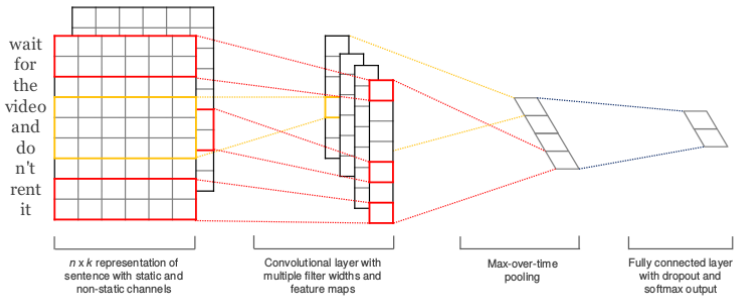


Fig. 2. Kim's architecture.

It begins by encoding all words into number indices. Each index corresponds to a word vector. Each training example will then be represented by a sequence of numbers. The sequence length will vary. Padding is therefore done into each sequence and for our purposes, the total number of numbers in a sequence is 59,

as in the paper. We then feed the sequence into the system. Each number will be looked up in the embedding layer and converted to a word vector and the whole sequence will form a matrix.

Feature maps are then created by convolving filters to the matrix and using a non-linearity after the convolution. There are three filter windows for this experiment - 3, 4, and 5. Each filter window has 100 feature maps. Note that the coefficients of the filters are initially random and then updated while training. The non-linearity used in our experiments is the *tanh* function.

After making a feature map, max over time pooling is performed. This means that from each feature map, only the maximum is recorded. Therefore, there will be a total of 300 features after max over time pooling is done. Then a dropout layer is added. Our dropout probability is 0.5. We finally add a softmax layer as the final layer with the weight vectors constrained to an l_2-norm.

Training is done through stochastic gradient descent over shuffled mini-batches with the Adadelta update rule where each mini-batch is made of 3000 examples. The dev set is comprised of 20% of the training set. We also kept the number of epochs to 30 and to provide for early stopping.

3.5 Model Variations

We experimented on two aspects. The first is dimension. We have two dimensions available - 100 and 300. Both are from skip-grams. Then we want to see if having word vectors fine-tuned would have an effect in the accuracy. This will be indicated by *CNN-static*, which just means that word vectors were taken from pre-trained word2vec but this was kept static. On the other hand *CNN-non-static* is the same as the first but it was tuned while training.

3.6 Evaluation

As stated earlier, we set aside 30% of the dataset for final evaluation. After the training is done, we apply the model on the tweets we set aside. After getting a prediction, we group the tweets that belong to the same user and get the majority prediction from all the tweets gathered for that user. We used the majority prediction as a final prediction for the user and base our accuracy off of that.

4 Results and Discussion

To recap, we did experiments on two different languages (English and Spanish), on two different classification tasks (age and gender), using two different vector dimensions (100 and 300), and two different treatments for the vectors (static and non-static). This gives a total of 16 experiments. Table 3 show the CNN accuracy results for English and Spanish. We can observe different patterns with these accuracy results.

We first look at the difference between accuracy over tweets and accuracy over users. The accuracy over tweets mean that each tweet is regarded as an example for which we evaluate the accuracy. Accuracy over users means that the tweets are agreggated first by the user who sent the tweet and uses the majority prediction as the final prediction for the user. The accuracy is evaluated on the user. We can see that using the majority predicted class for a tweet to predict the user generally improves the result except for Spanish gender evaluation. static word vectors with 300 dimensions.

Looking at the accuracies per user, we observe the effects of dimensionality as well as the effect of treating vectors static or non-static. The effect is different for each language on aspects of dimensionality. We can see that increasing the dimension generally improves the results for English. Only gender classification using static vectors decreases in accuracy. However it either lowers the accuracy or it does not have an effect on that of Spanish. One possible cause for this is the fact that the Spanish Wikipedia dump is significantly smaller in size than that of the English Wikipedia dump. This might make the resulting embeddings develop weaker relations to each other as compared to that of English.

The effect also varies when treating the vectors as static or non-static. The performance generally increases for both English and Spanish when moving from static to non-static. This is possibly because the vectors get finely tuned with more training data. There is an increase of 1.5% and 7.7% for English gender classification using 100 and 300 dimensions respectively. There's also a 1.3% increase for Spanish age classification using 100 dimensions while using 300 dimensions did not yield any difference. There's also an increase of 8.0% for Spanish gender classification using 300 dimensions. The other instances however lower the result when tuning the vectors. The biggest decrease is 4.0% which comes from gender classification using 100 dimensions.

Table 3. Accuracy comparison between evaluation by tweets and evaluation by user

	English				Spanish			
	Age		Gender		Age		Gender	
	static	non-static	static	non-static	static	non-static	static	non-static
100								
tweet	0.407	0.397	0.618	0.623	0.541	0.481	0.561	0.551
user	0.481	0.473	0.651	0.667	0.547	0.560	0.557	0.550
300								
tweet	0.410	0.409	0.626	0.613	0.538	0.467	0.693	0.653
user	0.496	0.473	0.643	0.721	0.547	0.547	0.507	0.587

We look at table 4 which shows the best result for convolutional neural networks that we were able to obtain against the best results from previous experiments. In the previous work [3], we have results comparing the accuracy of an

SVM classifier trained on *tfidf* against another SVM classifier trained on average of word vectors. In addition to what was done in the previous paper, we also experimented with skip-grams as well as varying the dimensions from 100 to 300 for skip-grams. The tasks and the dataset are the same and the test set was also the same for the previous experiment and the experiment detailed on this paper. The difference is that each training example uses all the tweets for each user in the previous experiment. This is different from this paper's experiment which uses each tweet as an example. The best setting has been with a convolutional neural network from looking at the results but there is no definitive architecture or vector dimensions that work best for everything. Convolutional neural network with 300 dimensions and static vectors work better on English age classification compared to an SVM with *tfidf* with an accuracy of 49.6% compared to 39.7%. Convolutional neural network with 300 dimensions and non-static vectors work better on English gender classification compared to an SVM trained on a 100 dimensional vector of averages trained on continuous bag of words. This has an accuracy of 72.1% compared to 70.2%. Convolutional neural network with 100 dimensions and non-static vectors work better on Spanish age classification compared to an SVM trained on either a 100 or 300 dimensional vector of averages trained on skip-grams. This has an accuracy of 56.0% compared to 54.7%. We equally have the same accuracy for Spanish gender classification - 69.3%. This is the same for convolutional neural networks with 100 dimension static vectors as well as SVM trained on a 300 dimensional vector of averages on skip-grams. It should also be noted that these comparisons are not definitive since the way each example is constructed for this experiment is different to that of previous work.

Table 4. Comparison between the best results of the CNN and previous work

	English		Spanish	
	age	gender	age	gender
CNN	**0.496**	**0.721**	**0.560**	**0.693**
past-work	0.397	0.702	0.547	**0.693**

Another aspect to look at is that the number of epochs for training. Looking at the learning rates from previous results, learning does not seem to plateau since the early stopping callback did not take into effect. We then ran the same experiment of age and gender classification on English and Spanish with 100 and 300 vector dimensions but only for static. The difference this time is that training was ran on 200 epochs to see if the accuracy would improve. The comparison is given in table 5. We can see that aside from Spanish gender classification, the accuracy improvements are marginal or none at all.

Table 5. Comparison of accuracy results based on different number of epochs for training

		English		Spanish	
		age	gender	age	gender
Dim=100					
	Epoch=30	0.481	0.651	0.547	0.557
	Epoch=200	0.496	0.636	0.547	0.640
Dim=300					
	Epoch=30	0.496	0.643	0.547	0.507
	Epoch=200	0.481	0.643	0.560	0.560

5 Conclusion and Recommendations

To summarize, we were able to use word vectors in conjunction to a convolutional neural network using Kim's architecture [9] as basis. We observed that using a bigger word vector dimension for English tasks improves the result but the same is not true for Spanish tasks. We observed that the effect also varies when tuning or not-tuning vectors. It generally hurts the performance when tuning except for English and Spanish gender that uses 300 dimensions when evaluating on users. There is a general tendency to have a better accuracy result for users instead tweets. And finally, we are able to report a better accuracy score for all four tasks as compared to previous results.

However this work has a lot of hyperparameters that were either fixed based on Kim's architecture or decided based on previous experiments. Some of it might be sub-optimal. For instance, we used $f(x) = tanh(x)$ as our activation function instead of $f(x) = relu(x)$ according to Kim's architecture. Another thing could be the number of feature maps. Sequence size is also an important parameter that was overlooked. This work was left to 59 based on Kim's work but the highest token count was 98. Another main concern is the use of vectors trained on wikipedia instead of twitter. Preprocessing also does not account for the fact that hyperlinks and twitter mentions be queried as a separate vector. This could give important information. Other things that could be experimented more are the number of layers, padding, the dropout, and even the regularization that was at the final layers. These are possible things to do for future work.

References

1. Miguel A Álvarez-Carmona, A Pastor López-Monroy, Manuel Montes-y Gómez, Luis Villaseñor-Pineda, and Hugo Jair-Escalante. Inaoe's participation at pan'15: Author profiling task. In *Working Notes Papers of the CLEF 2015 Evaluation Labs*, 2015.
2. Shlomo Argamon, Moshe Koppel, James W Pennebaker, and Jonathan Schler. Automatically profiling the author of an anonymous text. *Communications of the ACM*, 52(2):119–123, 2009.

3. Roy Bayot and Teresa Gonçalves. Author Profiling using SVMs and Word Embedding Averages—Notebook for PAN at CLEF 2016. In Balog et al. [23].
4. François Chollet. keras. `https://github.com/fchollet/keras`, 2015.
5. Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12(Aug):2493–2537, 2011.
6. Carlos E González-Gallardo, Azucena Montes, Gerardo Sierra, J Antonio Núñez-Juárez, Adolfo Jonathan Salinas-López, and Juan Ek. Tweets classification using corpus dependent tags, character and pos n-grams. In *Proceedings of CLEF*, 2015.
7. Michael Halliday, Christian MIM Matthiessen, and Christian Matthiessen. *An introduction to functional grammar*. Routledge, 2014.
8. Nal Kalchbrenner, Edward Grefenstette, Phil Blunsom, Dimitri Kartsaklis, Nal Kalchbrenner, Mehrnoosh Sadrzadeh, Nal Kalchbrenner, Phil Blunsom, Nal Kalchbrenner, and Phil Blunsom. A convolutional neural network for modelling sentences. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 212–217. Association for Computational Linguistics, 2014.
9. Yoon Kim. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*, 2014.
10. Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.
11. Xin Li and Dan Roth. Learning question classifiers. In *Proceedings of the 19th international conference on Computational linguistics-Volume 1*, pages 1–7. Association for Computational Linguistics, 2002.
12. Adrian Pastor Lopez-Monroy, Manuel Montes-y Gomez, Hugo Jair Escalante, Luis Villasenor-Pineda, and Esaú Villatoro-Tello. Inaoe's participation at pan'13: Author profiling task. In *CLEF 2013 Evaluation Labs and Workshop*, 2013.
13. Adrián Pastor López-Monroy, Manuel Montes-y Gómez, Hugo Jair Escalante, and Luis Villaseñor Pineda. Using intra-profile information for author profiling. In *CLEF (Working Notes)*, pages 1116–1120, 2014.
14. Suraj Maharjan, Prasha Shrestha, and Thamar Solorio. A simple approach to author profiling in mapreduce. In *CLEF (Working Notes)*, pages 1121–1128, 2014.
15. James Marquardt, Golnoosh Farnadi, Gayathri Vasudevan, Marie-Francine Moens, Sergio Davalos, Ankur Teredesai, and Martine De Cock. Age and gender identification in social media. *Proceedings of CLEF 2014 Evaluation Labs*, 2014.
16. Michał Meina, Karolina Brodzinska, Bartosz Celmer, Maja Czoków, Martyna Patera, Jakub Pezacki, and Mateusz Wilk. Ensemble-based classification for author profiling using various features. *Notebook Papers of CLEF*, 2013.
17. Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.
18. Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.
19. Bo Pang and Lillian Lee. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd annual meeting on Association for Computational Linguistics*, page 271. Association for Computational Linguistics, 2004.
20. Bo Pang and Lillian Lee. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Annual*

Meeting on Association for Computational Linguistics, pages 115–124. Association for Computational Linguistics, 2005.

21. Francisco Rangel, Paolo Rosso, Moshe Moshe Koppel, Efstathios Stamatatos, and Giacomo Inches. Overview of the author profiling task at pan 2013. In *CLEF Conference on Multilingual and Multimodal Information Access Evaluation*, pages 352–365. CELCT, 2013.
22. Francisco Rangel, Paolo Rosso, Martin Potthast, Benno Stein, and Walter Daelemans. Overview of the 3nd author profiling task at pan 2015. In L Cappellato, N Ferro, J Gareth, and E San Juan, editors, *CLEF 2015 Labs and Workshops, Notebook Papers*, volume 1391, 2015.
23. Francisco Rangel, Paolo Rosso, Ben Verhoeven, Walter Daelemans, Martin Pottast, and Benno Stein. Overview of the 4th Author Profiling Task at PAN 2016. In Krisztian Balog, Linda Cappellato, Nicola Ferro, and Craig Macdonald, editors, *Working Notes Papers of the CLEF 2015 Evaluation Labs*, volume 1609 of *CEUR Workshop Proceedings*, pages 750–784. CLEF and CEUR-WS.org, September 2016.
24. Francisco Rangel, Paolo Rosso, Ben Verhoeven, Walter Daelemans, Martin Potthast, and Benno Stein. Overview of the 4th Author Profiling Task at PAN 2016: Cross-genre Evaluations. In *Working Notes Papers of the CLEF 2016 Evaluation Labs*, CEUR Workshop Proceedings. CLEF and CEUR-WS.org, September 2016.
25. Radim Řehůřek and Petr Sojka. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta, May 2010. ELRA. `http://is.muni.cz/publication/884893/en`.
26. K Santosh, Romil Bansal, Mihir Shekhar, and Vasudeva Varma. Author profiling: Predicting age and gender from blogs. *Notebook Papers of CLEF*, 2013.
27. Jonathan Schler, Moshe Koppel, Shlomo Argamon, and James W Pennebaker. Effects of age and gender on blogging. In *AAAI Spring Symposium: Computational Approaches to Analyzing Weblogs*, volume 6, pages 199–205, 2006.
28. Yelong Shen, Xiaodong He, Jianfeng Gao, Li Deng, and Grégoire Mesnil. Learning semantic representations using convolutional neural networks for web search. In *Proceedings of the 23rd International Conference on World Wide Web*, pages 373–374. ACM, 2014.
29. Theano Development Team. Theano: A Python framework for fast computation of mathematical expressions. *arXiv e-prints*, abs/1605.02688, May 2016.
30. Julio Villena Román and José Carlos González Cristóbal. Daedalus at pan 2014: Guessing tweet author's gender and age, 2014.
31. Edson RD Weren, Viviane Pereira Moreira, and José Palazzo M de Oliveira. Exploring information retrieval features for author profiling. In *CLEF (Working Notes)*, pages 1164–1171, 2014.
32. Janyce Wiebe, Theresa Wilson, and Claire Cardie. Annotating expressions of opinions and emotions in language. *Language resources and evaluation*, 39(2-3):165–210, 2005.
33. Wen-tau Yih, Xiaodong He, and Christopher Meek. Semantic parsing for single-relation question answering. In *ACL (2)*, pages 643–648. Citeseer, 2014.

Análise de Algoritmos para Sistemas de Recomendação*

Nuno Miranda
d11797@uevora.pt

Universidade de Évora, Departamento de Informática
Orientador: Teresa Gonçalves

Resumo Os sistemas de recomendação são uma área bastante rica em abordagens e algoritmos de diferentes tipos. Quando se pretende trabalhar nesta área, é fundamental um levantamento prévio dessas principais abordagens e aproximações. Esse levantamento de abordagens de estado da arte já tinha sido elaborado do ponto de vista mais teórico e conceptual no âmbito de outros trabalhos. Neste trabalho pretendeu-se aprofundar a um nível mais prático, aplicando diversos algoritmos a um cenário real de recomendação para assim obter-se métricas para análise e comparação dos diversos algoritmos.

1 Introdução

Os sistemas de recomendação são técnicas e ferramentas de software que permitem dar sugestões sobre a escolha de um, ou vários itens, ao utilizador [12,34]. As áreas onde estes sistemas são mais utilizados são na recomendação de compras, músicas, filmes, destinos de férias, notícias e livros.

Os sistemas de recomendação, especialmente aplicados no contexto web e das novas tecnologias, permitiram a resolução da limitação conhecida como *fenómeno de cauda longa*[1] [43,13], que é quando apenas os itens de maior sucesso são apresentados, renegando sempre para segundo plano os restantes, mesmo que sejam mais apelativos para o utilizador. A aplicação dos mecanismos de recomendação em contextos web veio alterar esse comportamento, pois os itens exibidos não estão restringidos aos mais populares. Graças aos sistemas de recomendação, é possível apresentar aos utilizadores itens pouco populares, mas que ainda assim, preenchem as preferências de certos utilizadores.

A designação de item ou itens é normalmente utilizada para designar o que vai ser recomendado pelo sistema, assim como o conjunto de onde será extraída essa recomendação [34]. O termo de utilizador é por norma atribuído à pessoa que vai usufruir da recomendação do sistema [34]. No entanto não são apenas os itens que são analisados pelos sistemas de recomendação. Dependendo do sistema e da abordagem seguida, o próprio utilizador também pode ser analisado

* Este Artigo é destinado a Seminário III
[1] Do Inglês 'Long Tail Phenomenon'

durante esse processo [15,12,8]. As transações, são o termo frequentemente utilizado para designar as interações entre o utilizador e o sistema de recomendação. Interações essas, que fundamentalmente são a obtenção de dados que permitem aos algoritmos efetuar futuras recomendações [37].

A formalização do problema inerente aos mecanismos e sistemas de recomendação [4] passa por ter o utilizador c pertencente ao conjunto de utilizadores C e o item s pertencente ao conjunto de itens S com N elementos. Assim a função $U(c, s)$ é a responsável por obter a utilidade de uma recomendação. Para determinar o item ou itens com maior utilidade para um utilizador, é necessário aplicar essa função a todos os itens $U(c, s_1), ..., U(c, s_N)$. Após essa operação, podem-se obter os itens ordenados pela sua utilidade $s_{j_1}, ..., s_{j_N}$, ou apenas os K elementos mais relevantes $s_{j_1}, ..., s_{j_k}$ com $(K \leq N)$, ou ainda, apenas o item com maior utilidade, $s_j = \arg\max_{j \in S} U(c, j)$.

Para alcançar os itens com utilidade máxima existe um grande número de técnicas e de abordagens que são agrupadas em famílias por terem uma aproximação semelhante. Algumas utilizam técnicas de aprendizagem automática, teorias de aproximação e diversas heurísticas.

A função de utilidade nem sempre é obtida objetivamente para todos os itens, pois nem todos eles foram avaliados ou caracterizados de acordo com as preferências do utilizador. É sobretudo nesses itens não caracterizados que os sistemas de recomendação assumem uma importância elevada. Nesses casos a função de utilidade têm de ser estimada.

1.1 Abordagens de Filtragem de Conteúdos

Com esta abordagem, o sistema de recomendação gera os seus resultados de estimativa de utilidade, baseando-se nas características e atributos de outros itens que foram escolhidos ou preferidos pelo utilizador no passado. Ou seja, a obtenção da função de utilidade $U(c, s)$ para o utilizador c e para o item s é obtida através das utilidades dos outros itens para o mesmo utilizador $U(c, s_j)$, desde que os itens sejam todos semelhantes e com atributos comuns que permitam a comparação entre eles.

A aproximação baseada em conteúdos tem as suas origens em sistemas de extração de informação [6,36] e em sistemas de filtragem de informação [17]. Ambas as técnicas são bastante utilizadas porque muitos dos sistemas de recomendação, trabalham sobre itens com informação descritiva em língua natural, como é o caso da recomendação de páginas web, de notícias e de livros.

No entanto, os melhores sistemas de recomendação baseados em conteúdos não se ficam pelas técnicas de extração e de filtragem de informação. Frequentemente adicionam técnicas de criação de perfis. Essas abordagens obtêm dados para além do item em si, recolhendo informação sobre as preferências do utilizador. Esses dados extra, como já foi introduzido anteriormente, podem ser obtidos de forma implícita [30,5] ou explícita [29,30].

Nos sistemas que assentam em itens baseados em textos em língua natural, recorre-se muitas vezes à computação e extração de palavras-chave. As palavras-chave obtidas dos diversos textos acabam por ser os atributos de comparação

para a descoberta de itens que se enquadrem nos gostos do utilizador. Por exemplo o Sistema Fab [7] recomenda páginas Web ao utilizador tendo em conta as 100 palavras-chave mais importantes. De forma semelhante, o sistema Syskill & Webert [31] baseia as suas recomendações de páginas Web nas 128 palavras mais informativas. Para obter os termos mais significativos, existem várias aproximações, no entanto uma das mais utilizadas é o TFIDF[2] [17], que tem em conta a ocorrência do termo no documento em análise e no conjunto de todos os documentos de comparação do sistema de recomendação. Nesta técnica, um termo tem mais importância quanto mais se repete no documento, mas por sua vez, esse termo perde importância quantas mais vezes se repetir na coleção dos documentos.

1.2 Abordagens Colaborativas

Nos sistemas de recomendação baseados em abordagens colaborativas, a estimativa de utilidade sobre os diversos itens é efetuada a partir da análise das escolhas de outros utilizadores com perfil semelhante.

Uma das vantagens destas abordagens é permitir ter uma elevada, ou total, abstração sobre os itens e os seus atributos. Em situações onde os sistemas de recomendação baseados em conteúdos falham porque os itens têm poucos atributos ou esses atributos não são computáveis, esta limitação é completamente ultrapassada com os princípios das abordagens colaborativas [15,37]. Esta abordagem é uma das mais largamente utilizada na maioria das plataformas web que utilizam sistemas de recomendação com milhares ou milhões de itens e utilizadores, sendo até designada por correlação pessoa-pessoa [38].

Na construção de sistemas de recomendação deste tipo, e à semelhança dos sistemas de filtragem de conteúdos, também são criados vetores de atributos com pesos distintos para cada um deles, sendo posteriormente aplicadas diversas técnicas para correlacionar esses vetores. No entanto, neste caso, os atributos e respetivos valores são obtidos a partir de características dos utilizadores. A recolha dos atributos podem seguir uma aproximação explícita ou implícita (também à semelhança dos sistemas de filtragem de conteúdos).

Formalmente, esta abordagem baseia a sua recomendação na função utilidade para cada utilizador c e respetivo item s, sendo que a utilidade $U(c, s)$ é estimada através das utilidades $U(c_j, s)$, associadas ao item $s \in S$ e utilizadores $c_j \in C$ com características semelhantes a c.

O sistema Grundy [35] para a recomendação de livros, um dos primeiros a implementar esta abordagem, utilizava a designação de estereótipos para a aglutinação dos diferentes utilizadores em perfis, consoante as suas preferências. No entanto, a criação desses perfis era um processo inteiramente manual. Mais tarde, o sistema Tapestry [15], solicitava aos utilizadores que, manualmente, selecionassem outros utilizadores com gostos semelhantes, baseando-se no seu histórico de itens. Os sistemas GroupLens [21,33], Video Recommender [18] e Ringo [39] foram os primeiros a efetuar este tipo de operações automaticamente e a obter

[2] Sigla proveniente do Inglês, *Term Frequency-Inverse Document Frequency*

a respetiva recomendação automática. Mais tarde, surgiu uma nova geração de sistemas mais complexos e eficientes como o algoritmo de recomendação de livros da Amazon [26], o sistema PHOAKS [40] para a recomendação de informação relevante na Web e o sistema Jester [16] para a recomendação de anedotas online.

Os sistemas de recomendação colaborativos podem, segundo [10], ser agrupados em duas sub-classes tendo em consideração o seu funcionamento. A primeira classe, é baseada em memória ou heurísticas[3]; a segunda classe de algoritmos colaborativos é baseada em modelos[4].

1.3 Outras abordagens

Para além das abordagens mais comuns descritas anteriormente, existem outras aproximações que, sendo menos relevantes nos problemas de recomendação, acabam por ser bastante úteis quando utilizadas em conjunto com as técnicas principais (recomendação por filtragem de conteúdo e recomendação colaborativa). Da junção dessas técnicas, obtêm-se sistemas híbridos capazes de resolver algumas das limitações das abordagens principais [11].

A abordagem Demográfica [35] pode ser vista como um sub-tipo das técnicas colaborativas que baseiam a recomendação nos utilizadores semelhantes ao utilizador a que se pretende efetuar a recomendação. A grande diferença é que este não se baseia nos gostos e preferências dos outros utilizadores, mas sim em características pessoais dos utilizadores, como por exemplo, a idade, o estado civil, raça, género, habilitações literárias, trabalho, o país, região e cidade [32]. Embora esta abordagem só por si seja fraca e gere recomendações muito vagas e generalistas é uma técnica que, num sistema híbrido em associação com um sistema colaborativo, pode ajudar a resolver os problemas dos novos utilizadores e da fraca densidade de dados para utilizadores com gostos atípicos; nestes casos em que a recomendação não pode ser baseada na abordagem colaborativa, aplica-se a abordagem demográfica que, mesmo sendo mais genérica, consegue dar recomendações nesses casos específicos [32,4].

Outra abordagem que acaba por ser uma variação dos sistemas de filtragem de conteúdo é a filtragem baseada no contexto [3]. Esta concentra a sua análise nos atributos de contextualização e meta-informação sobre o item, ao invés de se focar nos seus atributos concretos. Um exemplo pode ser dado no contexto das notícias onde o sistema de recomendação, em vez de se focar na análise do conteúdo da notícia, baseia-se em quem escreveu a notícia, qual a categoria, a data de publicação ou até a organização que a publicou. Da mesma forma que um sistema híbrido com abordagem demográfica e a colaborativa permitia resolver alguns problemas da abordagem colaborativa, a filtragem de contexto num sistema híbrido com a filtragem de conteúdo permite resolver alguns problemas, nomeadamente o problema da entrada de um novo item no sistema.

---

[3] Do Inglês: *memory-based* ou *heuristic-based*
[4] Do Inglês: *model-based*

2 Ferramentas utilizadas

Para a realização deste trabalho e especialmente dos testes práticos foram utilizadas ferramentas tais como o MyMediaLite[24], Recommender Extension[27] e o Rapid Miner[2] que serão brevemente descritas de seguida.

Rapid Miner Para realizar os testes e comparações pretendidas foi utilizado o RapidMiner[2] que é uma plataforma integrada que permite efetuar testes e experiências de machine learning, data e text mining, análises preditivas e obtenção de business analytics. A versão utilizada foi a 5.3.015, a ultima versão gratuita antes do RapidMiner se tornar uma aplicação comercial.

Recommender Extension O Recommender Extension é como o nome indica uma extensão para o Rapid Miner, com um conjunto de algoritmos e ferramentas específicas para sistemas de recomendação. Esta extensão é desenvolvida pelo e-Lico, auto designado por "An e-Laboratory for Interdisciplinary Collaborative Research in Data Mining and Data-Intensive Science". Esta extensão, essencialmente faculta um conjunto de operadores para serem utilizados nos pipelines e workflows do Rapid Miner. Esses operadores estão agrupados em dois conjuntos, um de operadores preditivos de avaliação (Rating prediction operators) e o conjunto de operadores de recomendação de itens (Item recommendation operators)[27].

MyMediaLite O Recommender Extension, por sua vez, acaba por ser uma reimplementação da biblioteca MyMediaLite[24]. Uma biblioteca open source desenvolvida em C#/.NET e especializada em problemas de recomendação, nomeadamente em "rating prediction" e "item prediction from positive-only feedback"

3 Algoritmos

Durante a realização dos ensaios, foram utilizados diversos algoritmos para permitir posteriormente comparações entre a performance e resultados apresentados por cada um, e desta forma obter assim conclusões. De seguida será apresentada uma breve descrição dos diversos algoritmos utilizados.

- **Random** - Este algoritmo apenas atribui valores aleatoriamente para os seus resultados servirem de base de comparação com os resultados de outros algoritmos e permitir quantificar o ganho que esses algoritmos apresentam em relação a algo aleatório.
- **Item k-NN** - Algoritmo clássico para Sistemas de recomendação em que as avaliações são estimadas utilizando as avaliações anteriores dos k itens mais semelhantes ao que se pretende estimar. Matematicamente o algoritmo assenta sobre a correlação de Pearson e cosine similarity[42].

- **Slope One** - É um algoritmo para Sistemas de recomendação colaborativa introduzido em 2005. Destaca-se pela sua simplicidade nos algoritmos de recomendação colaborativa baseada nos itens. Nessa área é um dos mais simples de implementar e apresenta resultados semelhantes a algoritmos computacionalmente muito mais pesados e assenta em ponderação baseada em frequências[25].
- **Global Average** - Este algoritmo também faz parte da família de algoritmos colaborativos para estimar a classificação de itens. No entanto é bastante simplista e com resultados fracos comparativamente a algoritmos mais complexos, tendo apenas a vantagem de ser computacionalmente muito leve. Baseia-se simplesmente em calcular uma média de classificação e aplicar esse valor às estimativas a atribuir.
- **User Item Baseline** - Este algoritmo utiliza como ponto de partida o cálculo da média das avaliações para os diversos itens conhecidos e depois ajusta esse valor com um fator de inclinação ou tendência. Este algoritmo, ao contrário da maioria dos outros, permite aplicar várias iterações de otimização de parâmetros para obter o melhor resultado possível[23].
- **User k-NN** - Algoritmo clássico para Sistemas de recomendação em que as avaliações são estimadas utilizando as avaliações anteriores dos utilizadores k mais semelhantes ao que se pretende dar valores estimados. Matematicamente o algoritmo assenta sobre a correlação de Pearson e cosine similarity[42].
- **BP Slope One** - É um algoritmo para Sistemas de recomendação colaborativa introduzido em 2005. Destaca-se pela sua simplicidade nos algoritmos de recomendação colaborativa baseada nos itens. Nessa área é um dos mais simples de implementar e apresenta resultados semelhantes a algoritmos computacionalmente muito mais pesados e assenta em ponderação baseada em frequências bi-polares[25].
- **MF** - Este algoritmo também é designado por Matrix Factorization e como o nome indica efetua a factorização/decompozição de matrizes, em que uma matriz contem os dados dos utilizadores e outra os dados dos itens. Por vezes os valores de factorização excedem a dimensão do tipo numérico double originando problemas nos cálculos. Nesses casos utiliza-se o algoritmo seguinte BMF que tem um comportamento mais estável[28,22].
- **BMF** - é a abreviatura de Biased Matrix Factorization e é uma variante do algoritmo anterior em que existem regras explícitas para os valores dos atributos tanto dos utilizadores como dos itens[14]. Essas limitações permitem contornar os problemas de overflow do algoritmo MF (Matrix Factorization) e melhorar em certos cenários os resultados obtidos.
- **FWMF** - é a abreviatura de Factor Wise Matrix Factorization e é outra variante do algoritmo Matrix Factorization mas com mecanismos para prevenir overfitting e um conjunto de melhoramentos matemáticos para obter melhores resultados com melhor performance temporal e computacional, esses mecanismos podem ser observados em maior detalhe no artigo do próprio autor do algoritmo[9].

4 Dados

Os dados utilizados para realizar estas experiências incidiram todos sobre o mesmo domínio que são as preferências e avaliações de vídeos por parte de diversos utilizadores. Tendo uma grande amostragem nesse domínio é possível ter um ponto de partida e uma base de trabalho para a experimentação de diversos algoritmos de recomendação e assim observar os diferentes comportamentos das diversas abordagens referentes aos sistemas de classificação. Esses dados foram obtidos do GroupLens que é um laboratório do departamento de Ciência e Computação da Universidade do Minnestona. Esse laboratório é especializado em sistemas de recomendação, comunidades online e tecnologias ubíquas sempre com uma forte componente de computação social. O laboratório GroupLens disponibiliza uma ampla coleção de dados sobre as preferências de utilizadores no que diz respeito a filmes. Esses dados foram sendo recolhidos ao longo do tempo por via do MovieLens[1] , uma plataforma online que permite os utilizadores criarem uma conta, inserirem as suas preferências e receberem recomendações sobre novos filmes que serão do seu agrado. O próprio conjunto de dados de utilizadores e respetivas preferências vai crescendo ao longo do tempo à medida que novos filmes, utilizadores e preferências vão sendo adicionadas.

A sintaxe dos dados principais que dizem respeito a avaliação de utilizadores sobre diversos filmes apresentam a seguinte estrutura:

1::1246::4::978302091
2::1357::5::978298709

Embora a sua sintaxe tenha sido alterada para facilitar as etapas de processamento, nesta representação a primeira coluna indica o Id do utilizador, a segunda o Id do filme visualizado, a terceira coluna a avaliação atribuída e por fim o timestamp em que a avaliação foi atribuída. Para além do ficheiro com esta sintaxe de dados existem outros com informação complementar referente ao Id dos utilizadores e dos filmes.

Ao longo das experiências deste trabalho foram utilizados três conjuntos de dados do MovieLens, um com 100 mil recomendações, outro com 1 milhão e finalmente um com 10 milhões de recomendações. A tabela seguinte faz um resumo dos números envolvidos em cada um dos conjuntos.

Tabela 1. Quantificação dos conjuntos

	N Avaliações (1-5)	N Utilizadores	N Filmes
Conjunto 100K	100 mil	943	1682
Conjunto 1M	1 000 209	6040	3900
Conjunto 10M	10 000 054	71 567	10681

Nota: Cada utilizador avaliou pelo menos 20 filmes

5 Métricas utilizadas

Para fazer a avaliação do desempenho dos vários algoritmos utilizados durante as experiências, foram utilizadas três métricas principais, sendo elas:

Root Mean Square Error (RMSE) É uma medida que contempla as distâncias entre valores previstos por um modelo ou uma estimativa e os valores reais observados[20]. O RMSE é uma evolução do MSE (Mean Squared Error), o MSE também é uma média das distâncias entre os valores previstos e os valores reais, no entanto, essa distância é ao quadrado para assegurar que as distâncias negativas não anulam as distâncias positivas. No RMSE esses valores ficam em raiz quadrada de modo a ficarem na mesma ordem de grandeza que os valores originais de onde foram extraídas as distâncias.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} (f_i - y_i)^2}{n}}$$

Onde o f_i é o valor previsto e o y_i é o valor verdadeiro.

Mean Absolute Error (MAE) O Mean Absolute Error é uma medida estatística muito frequentemente utilizada para medir os desvios entre os valores previstos/estimados e os valores que realmente ocorreram[19]. Esta medida é obtida pela seguinte fórmula:

$$MAE = \frac{1}{n} \sum_{i=1}^{n} |f_i - y_i| = \frac{1}{n} \sum_{i=1}^{n} |e_i|$$

Esta métrica, como o nome sugere é uma média dos erros absolutos sendo estes:

$$|e_i| = |f_i - y_i|$$

Onde o f_i é o valor previsto e o y_i é o valor verdadeiro.

Normalized Mean Absolute Error (NMAE) O Normalized Mean Absolute Error como o nome indica é o valor da medida anterior (MAE) normalizada. Desta forma o valor obtido já não é absoluto e compreendido na escala em que os valores previstos e reais se encontram, mas sim, expresso em percentagem[41]. O seu cálculo a partir do MAE é bastante simples como pode ser observado na fórmula seguinte:

$$NMAE = \frac{MAE}{y_{max} - y_{min}}$$

Sendo y_{max} e y_{min} os valores máximos e mínimos, respetivamente, da escala em que os resultados e valores reais se encontram quantificados. Como nesta medida, o resultado é expresso em percentagem, isto permite que a análise dos valores obtidos seja abstraída dos valores e intervalos utilizados.

6 Testes e Resultados

Para efetuar testes sobre a eficácia dos diversos algoritmos, os conjuntos de dados originais foram partidos em conjuntos de treino e conjuntos de testes. Tanto para o conjunto de 100 mil, 1 milhão e 10 milhões de avaliações, foram extraídas um quarto (1/4) das avaliações dos primeiros 100 utilizadores para servirem de conjunto de testes. O conjunto de testes criado tinha então as diversas avaliações que iriam ser estimadas pelos algoritmos de recomendação. Como nesses dados de testes, era conhecida a avaliação real atribuída pelo utilizador, seria fácil posteriormente, obter métricas de performance e de desvio entre a recomendação e a avaliação realmente atribuída.

Após ter sido efetuada uma fase de pré-processamento para obter esta divisão do conjunto de dados originais para os respetivos conjuntos de treino e de testes, efetuou-se a experiencia de aplicar todos os algoritmos disponíveis no Recomendation Extension do RapidMiner, aos diferentes pares de conjuntos treino/teste e assim observar como estes se comportariam com diferentes quantidades de dados, também permitiria observar as diferenças de performance entre os diversos algoritmos para os mesmos conjuntos de treino/teste.

Para realizar diversos testes (diferentes conjuntos de dados Vs diversos algoritmos) foi criado um pipeline no Rapidminer para assim se definir um workflow dos testes igual para todos os diferentes conjuntos em teste. No caso do conjunto de 10M não foi possível correr alguns dos algoritmos. Tal problema deveu-se à necessidade de uma grande quantidade de memória para processar os 10 milhões de classificações presentes nesse conjunto de dados. Os algoritmos com esse problema foram o Item k-NN, BP Slope One, User k-NM e Slope One. Tirando esse problema pontual, num total de 30 experiências previstas, conseguiram-se realizar com sucesso 26.

Tabela 2. Tabela de Resultados

	100K			**1M**			**10M**		
	RMSE	**MAE**	**NMAE**	**RMSE**	**MAE**	**NMAE**	**RMSE**	**MAE**	**NMAE**
Random	*2,154*	*1,770*	*0,442*	*2,145*	*1,760*	*0,440*	*2,131*	*1,730*	*0,433*
Item k-NN	0,914	0,721	**0,180**	0,893	0,698	0,174	—	—	—
FWMF	0,963	0,748	0,187	0,876	0,676	**0,169**	0,765	0,574	**0,143**
BMF	0,973	0,766	0,192	0,867	0,681	**0,170**	0,749	0,575	**0,144**
MF	0,954	0,749	0,187	0,859	0,671	**0,168**	0,760	0,572	**0,143**
BP Slope One	0,949	0,730	**0,182**	0,951	0,722	0,181	—	—	—
User k-NN	0,926	0,733	**0,183**	0,901	0,702	0,176	—	—	—
user item Baseline	0,945	0,756	0,189	0,914	0,718	0,179	0,826	0,637	0,159
Global Average	1,144	0,966	0,241	1,103	0,930	0,232	1,050	0,874	0,219
Slope One	0,921	0,727	**0,182**	0,908	0,710	0,177	—	—	—

7 Conclusões e Trabalho Futuro

Os resultados estão divididos em três tabelas, cada uma representando a mesma experiência sobre os três conjuntos de dados (100k, 1M, 10M). Das três métricas registadas, a que foi utilizada para obter a maioria das conclusões foi a Normalized Mean Absolute Error (NMAE) por ser a única que os resultados são normalizados e expressos entre 0 e 1, permitindo uma maior abstração sobre os valores absolutos devolvidos pelas outras medidas. Em cada tabela, para a medida do NMAE, foram assinaladas a negrito os três melhores valores registados. Em modo de resumo podem-se extrair diversas conclusões, nomeadamente:

- Os resultados melhoraram ligeiramente, à medida que o conjunto aumentou de dimensão (100K $\Rightarrow$ 1M $\Rightarrow$ 10M)
- Os algoritmos com melhores resultados no conjunto de 100K são diferentes que os melhores algoritmos do conjunto 1M e 10M.
- O conjunto 1M e 10M partilham os algoritmos com os melhores resultados.
- Sendo o algoritmo Random um baseline para comparações, é possível observar que para todos os conjuntos (100K, 1M, 10M) e que para todos os restantes algoritmos utilizados, houve um ganho bastante assinalável comparativamente ao Random, desta forma conclui-se que os algoritmos tem um ganho assinalável comparativamente a algo aleatório.
- O algoritmo Global Average, devido ao seu funcionamento extremamente simplista, ainda assim apresenta resultados melhores que o algoritmo Random, no entanto tem piores resultados que todos os outros algoritmos mais complexos envolvidos nos testes.

Trabalho futuro Já durante a parte final da realização deste trabalho, o GroupLens disponibilizou outro conjunto de dados sobre avaliações de filmes, sendo um conjunto superior aos conjuntos disponibilizados anteriormente e utilizados nestes ensaios (100K, 1M, 10M). Esse novo conjunto de dados tem um total de 100 milhões de avaliações (100M). Seria interessante que com recurso a hardware com maiores capacidades de memória e de processamento fossem replicadas estas experiências a esse conjunto de dados. Outra experiência interessante de realizar futuramente, seria a introdução de atributos de meta-dados sobre os itens e/ou atributos demográficos e observar o impacto dessas alterações sobre os resultados obtidos nestes ensaios.

Referências

1. Grouplens - social computing research at the university of minnesota (2016), `http://grouplens.org/datasets/movielens/`
2. Rapidminer - open source predictive analytics (2016), `https://rapidminer.com/`
3. Abowd, G.D., Dey, A.K., Brown, P.J., Davies, N., Smith, M., Steggles, P.: Towards a better understanding of context and context-awareness. In: Proceedings of the 1st International Symposium on Handheld and Ubiquitous Computing. pp. 304–307. HUC '99, Springer-Verlag, London, UK, UK (1999), `http://dl.acm.org/citation.cfm?id=647985.743843`

4. Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. IEEE Trans. on Knowl. and Data Eng. 17(6), 734–749 (Jun 2005), http://dx.doi.org/10.1109/TKDE.2005.99
5. Anand, S.S., Mobasher, B.: Intelligent techniques for web personalization. In: Proceedings of the 2003 International Conference on Intelligent Techniques for Web Personalization. pp. 1–36. ITWP'03, Springer-Verlag, Berlin, Heidelberg (2005), http://dx.doi.org/10.1007/11577935_1
6. Baeza-Yates, R.A., Ribeiro-Neto, B.: Modern Information Retrieval. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA (1999)
7. Balabanović, M., Shoham, Y.: Fab: Content-based, collaborative recommendation. Commun. ACM 40(3), 66–72 (Mar 1997), http://doi.acm.org/10.1145/245108.245124
8. Basilico, J., Hofmann, T.: Unifying collaborative and content-based filtering. In: In ICML. pp. 65–72. ACM Press (2004)
9. Bell, Y.K.A.C.V.R.: Modeling relationships at multiple scales to improve accuracy of large recommender systems. Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD '07), ACM, New York, USA (2007)
10. Breese, J.S., Heckerman, D., Kadie, C.: Empirical analysis of predictive algorithms for collaborative filtering. In: Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence. pp. 43–52. UAI'98, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1998), http://dl.acm.org/citation.cfm?id=2074094.2074100
11. Burke, R.: Hybrid recommender systems: Survey and experiments. User Modeling and User-Adapted Interaction 12(4), 331–370 (Nov 2002), http://dx.doi.org/10.1023/A:1021240730564
12. Burke, R.: The adaptive web. chap. Hybrid Web Recommender Systems, pp. 377–408. Springer-Verlag, Berlin, Heidelberg (2007), http://dl.acm.org/citation.cfm?id=1768197.1768211
13. Celma, Ò.: Music Recommendation and Discovery in the Long Tail. Ph.D. thesis, Universitat Pompeu Fabra, Barcelona (2008), static/media/PhD_ocelma.pdf
14. Gemulla, E.N.P.J.H.A.Y.S.R.: Large-scale matrix factorization with distributed stochastic gradient descent. Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD '11), ACM, New York, USA (2011)
15. Goldberg, D., Nichols, D., Oki, B.M., Terry, D.: Using collaborative filtering to weave an information tapestry. Communications of the ACM 35, 61–70 (1992)
16. Goldberg, K., Roeder, T., Gupta, D., Perkins, C.: Eigentaste: A constant time collaborative filtering algorithm. Inf. Retr. 4(2), 133–151 (Jul 2001), http://dx.doi.org/10.1023/A:1011419012209
17. Hanani, U., Shapira, B., Shoval, P.: Information filtering: Overview of issues, research and systems. User Modeling and User-Adapted Interaction 11(3), 203–259 (Aug 2001), http://dx.doi.org/10.1023/A:1011196000674
18. Hill, W., Stead, L., Rosenstein, M., Furnas, G.: Recommending and evaluating choices in a virtual community of use. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. pp. 194–201. CHI '95, ACM Press/Addison-Wesley Publishing Co., New York, NY, USA (1995), http://dx.doi.org/10.1145/223904.223929

19. Hyndman, A.B.K.R.J.: Another look at measures of forecast accuracy. International Journal of Forecasting, Volume 22, Issue 4 Pages 679-688(ISSN 0169-2070) (oct 2006)
20. J., C.F.A.: Error measures for generalizing about forecasting methods: Empirical comparisons. International Journal of Forecasting, 8, issue 1, p. 69-80 (1992)
21. Konstan, J.A., Miller, B.N., Maltz, D., Herlocker, J.L., Gordon, L.R., Riedl, J.: Grouplens: Applying collaborative filtering to usenet news. Commun. ACM 40(3), 77–87 (Mar 1997), http://doi.acm.org/10.1145/245108.245126
22. Koren, Y.: Factorization meets the neighborhood: a multifaceted collaborative filtering model. Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD '08), ACM, New York, USA (2008)
23. Koren, Y.: Factor in the neighbors: Scalable and accurate collaborative filtering. ACM Trans. Knowl. Discov. Data 4, 1, Article 1, New York, USA (dec 2010)
24. a. S. R. a. C. F. a. L. Schmidt-Thieme, Z.G.: Mymedialite: A free recommender system library. Proceedings of the 5th ACM Conference on Recommender Systems (RecSys 2011) (2011)
25. Lemire, A.M.D.: Slope one predictors for online rating-based collaborative filtering. Proceedings of the 2005 SIAM International Conference on Data Mining (2005)
26. Linden, G., Smith, B., York, J.: Amazon.com recommendations: Item-to-item collaborative filtering. IEEE Internet Computing 7(1), 76–80 (Jan 2003), http://dx.doi.org/10.1109/MIC.2003.1167344
27. M., A.F.N.B.M.S.T.M.: Extending rapidminer with recommender systems algorithms. In: RapidMiner Community Meeting and Conference. Budapest, Hungary (2012)
28. M., R.S.N.S.J.D.: Fast maximum margin matrix factorization for collaborative prediction. Proceedings of the 22nd international conference on Machine learning (ICML '05), ACM, New York, USA, (2005)
29. Mahmood, T., Ricci, F.: Improving recommender systems with adaptive conversational strategies. In: Proceedings of the 20th ACM Conference on Hypertext and Hypermedia. pp. 73–82. HT '09, ACM, New York, NY, USA (2009), http://doi.acm.org/10.1145/1557914.1557930
30. McSherry, F., Mironov, I.: Differentially private recommender systems: Building privacy into the net. In: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 627–636. KDD '09, ACM, New York, NY, USA (2009), http://doi.acm.org/10.1145/1557019.1557090
31. Pazzani, M., Billsus, D.: Learning and revising user profiles: The identification ofinteresting web sites. Mach. Learn. 27(3), 313–331 (Jun 1997), http://dx.doi.org/10.1023/A:1007369909943
32. Pazzani, M.J.: A framework for collaborative, content-based and demographic filtering. Artif. Intell. Rev. 13(5-6), 393–408 (Dec 1999), http://dx.doi.org/10.1023/A:1006544522159
33. Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., Riedl, J.: Grouplens: An open architecture for collaborative filtering of netnews. In: Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work. pp. 175–186. CSCW '94, ACM, New York, NY, USA (1994), http://doi.acm.org/10.1145/192844.192905
34. Resnick, P., Varian, H.R.: Recommender systems. Commun. ACM 40(3), 56–58 (Mar 1997), http://doi.acm.org/10.1145/245108.245121
35. Rich, E.: Readings in intelligent user interfaces. chap. User Modeling via Stereotypes, pp. 329–342. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1998), http://dl.acm.org/citation.cfm?id=286013.286035

36. Salton, G. (ed.): Automatic Text Processing. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA (1988)
37. Schafer, J.B., Frankowski, D., Herlocker, J., Sen, S.: The adaptive web. chap. Collaborative Filtering Recommender Systems, pp. 291–324. Springer-Verlag, Berlin, Heidelberg (2007), http://dl.acm.org/citation.cfm?id=1768197.1768208
38. Schafer, J.B., Konstan, J.A., Riedl, J.: E-commerce recommendation applications. Data Min. Knowl. Discov. 5(1-2), 115–153 (Jan 2001), http://dx.doi.org/10.1023/A:1009804230409
39. Shardanand, U., Maes, P.: Social information filtering: Algorithms for automating word of mouth. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. pp. 210–217. CHI '95, ACM Press/Addison-Wesley Publishing Co., New York, NY, USA (1995), http://dx.doi.org/10.1145/223904.223931
40. Terveen, L., Hill, W., Amento, B., McDonald, D., Creter, J.: Phoaks: A system for sharing recommendations. Commun. ACM 40(3), 59–62 (Mar 1997), http://doi.acm.org/10.1145/245108.245122
41. Wang, A.C.B.Z.: Mean squared error: Love it or leave it? a new look at signal fidelity measures. IEEE Signal Processing Magazine, vol. 26, no. 1, pp. 98-117 (jan 2009)
42. Wen, Z.: Recommendation system based on collaborative filtering (2008)
43. Yin, H., Cui, B., Li, J., Yao, J., Chen, C.: Challenging the long tail recommendation. CoRR abs/1205.6700 (2012), http://arxiv.org/abs/1205.6700

Efficient Fingers Calibration Technique for Braille Touchscreen System

Puthnith Var, Luís Rato, and Miguel Barão

Department of Informatics
University of Évora
puthnith@gmail.com, lmr@uevora.pt, mjsb@uevora.pt

Abstract. Touchscreen devices, such as tablets and smartphones, have started to replace computers and laptops because of its gestures ability even though button-press-liked touch sensation is absent. Users can compose or note down messages and their ideas easily. However, the touch sensation is an obstacle for visually impaired people. This paper shows the first part of our proposed technique to address the problem and provide the original Braille composition to touchscreen devices. Besides providing fixed areas assigned to all the fingers on the screen, our efficient fingers calibration technique allows users to type on any area regardless to the position of each finger. As a result, users can freely compose texts using Braille system with any direction and composition of their hands.

Keywords: braille, touchscreen, fingers, calibration

1 Introduction

Touching on screens virtual keyboard provides different sensations and natural feelings than pressing buttons on computer's keyboards. On a computer's keyboard, the users are able to rest their fingers during typing and it allows them to recognize buttons staying under their fingers along with the sense of knowing a key is pressed or being pressed. However, this type of feeling, the touch-and-press action, is not possible to obtain while typing or touching on touchscreen devices despite the fact that it is being touched or pressed lightly or heavily. Typically, the users are required to look at the device's screens to know whether they touch the right area of the virtual buttons.

After all, the touchscreen devices discourage people who are visual impaired. When they touch the screen, they cannot track the location of the virtual buttons. Even though some devices provide responding voice which is not much comfortable, the devices create problem of not letting users to express their own thoughts and feedbacks as written texts. They, however, can record their voices but doing it in public places, such as on buses or in trains, disturbs other people and also loses their privacy. Thus, the devices' system has to be more intelligent to detect the fingers location to provide eyes-free text composition and it is strongly required for the visual impaired people.

People with visual impairments use Braille characters [1] to read by touching and they compose with a device called Braille keyboard. In order to compose on a touchscreen device, they need an extended Braille keyboard hardware that is connected to the device via the Bluetooth technology or a cable. The way they read or understand on the device is different from examining by touch on a surface of rough braille letters. They listen to a responded voice or audio [2][3] from the device to get information by touching on the screen.

In our previous work [4], we solve the problem of identifying all the touch-points on touchscreen devices. In the initialization or calibration stage, a user is required to put eight fingers on the touchscreen area of a touchscreen device. The system analyzes the provided touch-points or touched locations and responses back to the user in least than a second of which touch-points are the index, middle, ring, and little fingers of the user's left and right hands. However, what we have done is limited to different hands compositions. Users are required to put all the fingers in a horizontal or vertical line composition rather than in parallel or crossed hands compositions.

In this paper, we introduce a new technique to improve the calibration stage which addresses the limitation of the previous work and allows users to freely put their fingers in any form in order to compose Braille characters.

2 Braille Characters

Braille is a system that enables blind and visually impaired people to read and write through touch. Louis Braille invented Braille in 1821 and it consists of raised dots arranged in cells. A cell is made up of six dots that fit under the fingertips, arranged in two columns of three dots each as shown in Fig. 1. Each cell represents a letter, a word, a combination of letters, a numeral or a punctuation mark [5].

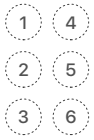
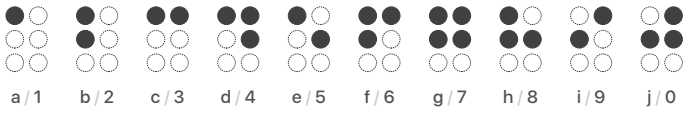


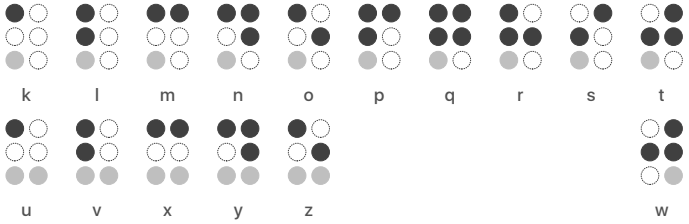
Fig. 1: The Braille cell

In this section, we would like to give an introduction to Braille characters to know how visual impaired people understand the language. Basically, there are 10 unique patterns composing from four distinguished points represented 10 numeric digits and the first ten alphabets (A to J) using the top four dots (1, 2, 4, 5).

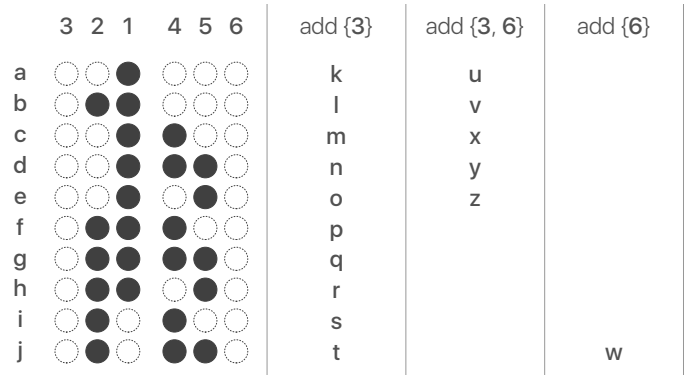
In Fig. 2a, there are 1 symbol (character or number) represented by 1 dot, 4 symbols represented by 2 dots, 4 symbols represented by 3 dots, and 1 symbol



(a) The 10 basic Braille patterns: characters A to J, number 1 to 9, 0



(b) The other 16 Braille patterns which using the same (a) patterns



(c) The Braille patterns for composition

Fig. 2: Braille characters and their patterns

represented by 4 dots. there are only 10 combinations out of 15. Other alphabets are the combination of the 10 patterns with the 2 bottom dots as shown below.

To compose the next ten alphabets (K to T), the 3rd dot is added to the 10 patterns as shown in Fig. 2b. However, while the U, V, X, Y, Z characters use 2 bottom dots (3rd and 6th dots) with the first 5 patterns, W character is shifted to 10th pattern with only 1 dot (6th dot). The odd of alphabet W is a bit confusing at first but this is because the French alphabet did not have that letter or not used frequently when Louis Braille invented it.

To compose the 26 alphabets, 6 braille dots are required. The patterns is shown in Fig. 2a and 2b are used for reading where visual impaired people use their fingertips to touch and recognize the cells. For Braille composition as shown in 2c, however, those 6 dots are laid out in a horizontal line where the first (dot

3, 2, and 1) and second (dot 4, 5, and 6) columns represented the left and right hands, respectively.

The Fig. 2c shows how the cells or patterns look like if we lay them out in a horizontal line. Dots 3, 2 and 1 is for the ring, middle and index fingers of the left hand, respectively. Dots 4, 5, and 6 is for the ring, middle and index fingers of the right hand, respectably. The dot's positions show the composition of characters from A to J. Characters K to J can be composed by add *dot 3* to the compositions. Other characters is composed as shown in other columns.

3 Calibration

3.1 Naive Algorithm

In the fundamental of fingers calibration stage, most of Braille systems requires users to input all their eight or six fingers on touchscreen devices to calibrate or identify each finger's location. However, this calibration stage is optional for some applications such as the official Braille inputs system for iOS devices. The iOS Braille inputs automatically defines the regions of fingers for users, left and right hands on the left and right areas of devices, respectively. This arrangement sometimes gives a restriction to users for not freely forming their fingers in different ways.

The naive algorithm [4] of identifying all the eight fingers is described as follow. A user is required to place or input all eight fingers on a device. Usually the eight fingers or touch-points is laid down horizontally from the left to right edges of the device. It is noticed that when the user inputs the fingers simultaneously, a returned collection of touch-points is a set of unordered touch-points. Thus, we cannot say the first touch-point is the first finger and so on. In this case, the touch-points are required to be sorted ascending of x-axis where the first element (the first touch-point in the set) is the first finger, the last element is the last finger, and so on. As we can see from this approach, a set of ordered touch-points is required in order to identify each given touch-point. This approach can only apply to around 90% of times that users put their fingers horizontally and restrictedly to only tablet devices.

This paper explains how we would like to overcome the arrangement and eight fingers calibration problems and provide many possible ways of composing braille characters.

3.2 Problem and Limitation

Some other cases, users may put their hands one on top of another (in case of small screen devices) or vertically parallel or overlapping each other, which the naive algorithm cannot tell the left and right hands. Thus, identifying the fingers is impossible.

Inputing all eight fingers together on touchscreen devices is limited to only tablet devices. That is because most tablet devices can detect up to 11 simultaneously touches at once, and smart phone devices allow only 5 simultaneously

touches. Thus, the calibration with eight fingers is impossible for small screen devices.

3.3 Proposed Technique

We want to support all kinds of touchscreen devices. Thus, the proposed technique requires users to provide four different inputs or Braille characters.

1. Braille Character A – *Left Index* finger
2. Braille Character B – *Left Index* and *Left Middle* fingers
3. Braille Character C – *Left Index* and *Right Index* fingers
4. Braille Character D – *Left Index*, *Right Index*, and *Right Middle* fingers

The four inputs sequence looks simple but it provides many parameters which help the system to predict the other four fingers, the right and little fingers of the left and right hands. The four inputs sequence, A-B-C-D, is not accidentally chosen. We have been studied closely to Braille characters and here we explain the reason of the inputs sequence. There are 3 parameters that we need to identify from the four inputs sequence: the left and right hands, the gap between fingers, and the slope of each hand.

Hands Identification We know that in order to compose a Braille character, users are required to use 2 hands explicitly; obviously, they are left and right hands. If we look at the Braille patterns, character A is the easiest and only one character that is composed by a single dot, the 1st dot, or the left index finger. From this single input, we can see that the left hand is located in a circle with the input in the middle. Thus, character A is chosen to identify the left hand.

To identify the right hand, character C and E are the candidates. Why? To avoid the problem that users may input different fingers at the same location, we need a reference to the first input, character A (or the 1st dot). Thus, the second input requires users input the left index finger with one of the fingers from the right hand. Braille character C is the combination of left and right index fingers. Braille character E is the combination of left index and right middle fingers. These characters are the only 2 inputs with 2 dots and have a reference to the 1st dot. Because character C is the foremost character and easier to compose compared to character E, character C is chosen to identify the right hand.

By the inputs of character A and C, we can identify the left and right hands and accurately 1st (left index) and 4th (right index) dots of the Braille cell. However, we cannot be sure where the other fingers are located.

Fingers' Gap and Slope Identifying other fingers seems to be easy if we assume users use tablets, which other fingers would be laid out to the left and right side of the 1st and 4th dots, respectively. However, there are more cases such as the hands can be parallel each other or the hands' size and shape. Thus, how to predict the location of other fingers is unclear with just the sequence of the two inputs.

Beside identifying hands, we require more inputs to identify the fingers' gap and slope. Thus, the position of the 2nd (left middle) and 5th (right middle) dots have to be clearly identified. The best candidate to identify 2nd dot is character B because it has a reference dot to the 1st dot. To find the 5th dot, character E and D can be the candidates which both of them have the 1st dot. However, character D has also the 4th dot which is stronger to identify the 5th dot.

With the input sequence, A-B-C-D, we can find the exact positions for the 1st, 2nd, 4th, and 5th dots. With those four positions, we can calculate the gaps and slopes between the 1st–2nd dots and the 4th–5th dots for the left and right hands, respectively, in order to find other four positions for ring and little fingers.

We would like to point out that when a user try to input the same location of a dot or a finger, the location of the finger will not be the same. Thus, a distance constraint is required. The distance is a radius from the first given location. Due to the movement of the user's hands, if the second input of the user's finger is not in a defined area from the first input, the user is required to give the second input again. Note that the movement of a user's fingers will be addressed in our next paper.

4 Results

The proposed calibration technique has been tested on iOS touchscreen devices and the results are accurate as expected. Users are able to freely compose Braille character in any hands compositions. Fig. 3 shows number of possible hands compositions. Users can compose Braille character in parallel, stack, intersection, opposition, and any other hands compositions as shown in the figure.

5 Conclusion

We can see that touchscreen devices provide a lot of advantages and functionalities to aid people in many different ways. However, the devices cannot fulfill the requirement of people with visual impairments because of the lack of pressing buttons sensation. Thus, we challenge the obstacle and would like to encourage those people to use touchscreen devices as notebooks like other people.

We provide the first step toward a novel Braille touchscreen system. We propose a new calibration technique which works on both small and big touchscreen sizes. Because small touchscreen devices such as smart phones allow only 5 simultaneously touches, which the eight inputs calibration does not work. Our proposed technique requires only one or two simultaneously touches with four inputs sequence, as simple as typing A-B-C-D. For visual impaired people, those first four Braille characters or compositions are at their fingertips.

With input Braille characters A to D, we can easily identify all the eight fingers by studying the locations, gaps and slopes of the inputs. As a result, users can compose Braille characters regardless to any direction and composition of their hands.

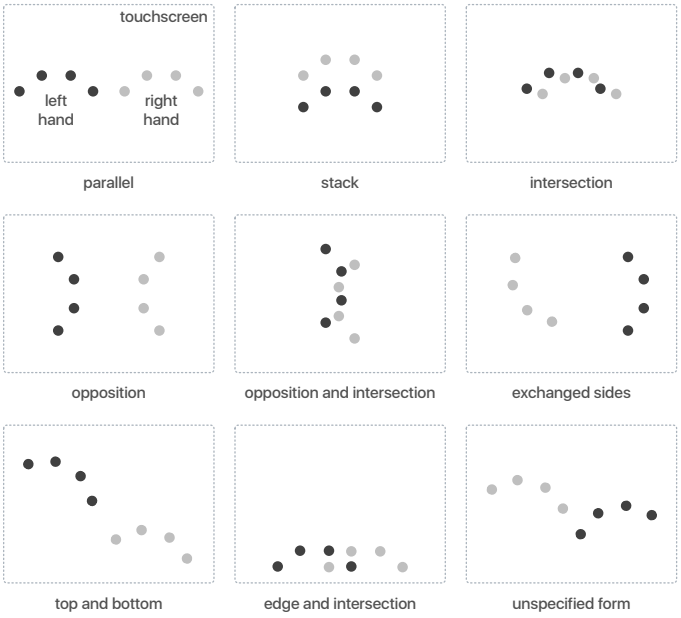


Fig. 3: Possible hands compositions

Finally, we expected to provide a perfect Braille touchscreen system to people with visual impairments. We hope that the output of our research gives advantages to not only the people with visual impairments but also to researchers and developers to contribute and build a better world together.

References

1. "Procedure for writing words, music and plain song using dots for the use of the blind and made available to them," *Royal Institution Of Blind Youth*, Paris, 1829.
2. Speech Enabled Eyes Free Android Applications http://code.google.com/p/ eyes-free/
3. VoiceOver http://www.apple.com/accessibility/iphone/vision.html
4. Var, P., Gonçalves T., Barão. M., "An Effective Fingers Detection Method for Braille Touchscreen Keyboard on Tablet Devices," *The 5th Workshop in Informatics of the University of Évora (JIUE 2015)*, Évora, 2015.
5. Braille Basics http://www.brailleauthority.org/learn/braillebasic.pdf

Aplicação da Ontologia PROV-O ao crime de branqueamento de capitais

Gonçalo J. F. Carnaz[1] and Carlos P. Caldeira[1]

Universidade de Évora, Departamento de Informática,Évora,
d34707@alunos.uevora.pt, ccaldeira@di.uevora.pt

Resumo Actualmente, os investigadores criminais deparam-se com novos desafios, entre eles: o combate ao branqueamento de capitais. Dado a enorme quantidade de dados que são recolhidos e processados durante as investigações, a necessidade de introdução de sistemas computacionais que permitem a representação do conhecimento do dominio em questão, para que depois possam ser processados. Assim, as ontologias surgem como forma de representação de conhecimento existente em diversos dominios. Assim, a ontologia PROV-O pode-se adaptar à representação do conhecimento criminal?.

Keywords: lavagem de dinheiro, ontologias, relatórios policiais.

1 Introdução

O branqueamento de capitais é um dos crimes mais comuns nos nossos dias, sendo um dos crimes mais difíceis de investigar por parte das polícias e ministério publico, dado a quantidade de intervenientes e evidências geradas. Assim, a representação deste conhecimento através de uma ontologia pode ser um passo na ajuda no combate a este tipo de crime. O que pretendemos neste artigo é a possibilidade da ontologia PROV-O representar o conhecimento inerente aos crimes em questão, ou servir de suporte a uma framework que possa ajudar o investigador na detecção de crimes de branqueamento de capitais. Na secção 2 descrevemos os estudos realizados acerca da aplicação das ontologias ao crime financeiro, sobretudo ao branqueamento de capitais. Na secção 3 explicamos resumidamente a ontologia Prov-O, descrevemos o crime de branqueamento de capitais na secção 4, seguidamente utilizamos a ontologia Prov-O para representar o conhecimento associado ao crime de branqueamento de capitais. Finalmente na secção 6 elaboramos as nossas conclusões.

2 Estudos prévios

Nas últimas décadas, diversos investigadores dedicaram o seu trabalho à representação do conhecimento em diferentes domínios: medicina, ensino, bibli-

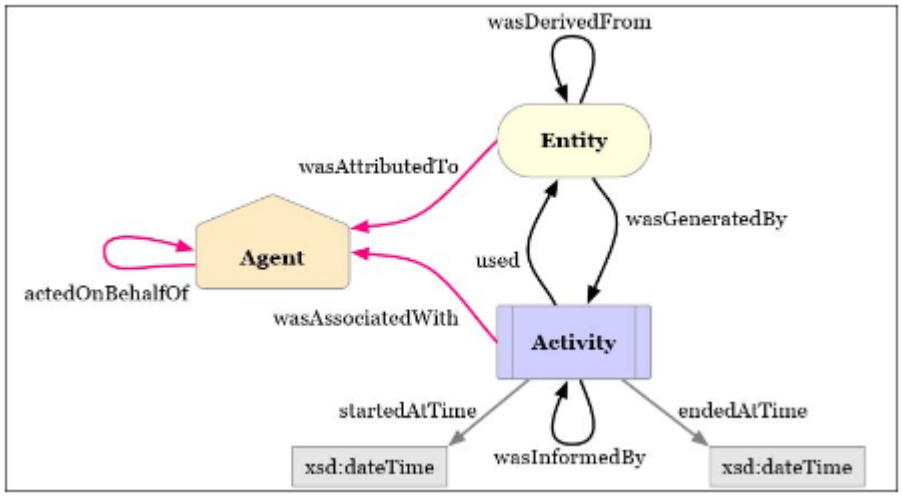
otecas, etc,. Existem alguns estudos relacionados com ontologias e o crime de branqueamento de capitais ou fraude financeira, que pretendem representar o conhecimento inerente a este domínio. Nos próximos parágrafos passamos a resumir o estado da arte associado às ontologias ligadas aos crimes financeiros. No artigo [5], os autores definem os passos para uma ontologia, com o nome FF POIROT, que representa o conhecimento no domínio do crime financeiro. Em 2014, os autores do artigo [10] procuraram representar as transações financeiras suspeitas através de sistema pericial, baseado numa ontologia e num conjunto de regras. Seguindo as regras de desenho aplicado a ontologias, foi criado pelos autores do artigo um conjunto de classes, objetos e propriedades que representam as transações a serem processadas pelo sistema pericial. Adicionalmente, um conjunto de regras, usando SWRL (Semantic Web Rules Language), por forma a inferir novo conhecimento através do conhecimento existente. Em [7], os autores definem uma ontologia que possa mapear o conhecimento inerente aos crimes que estamos a analisar, construindo uma ontologia que pode ajudar a descobrir esquemas de lavagem de dinheiro. São definidas entidades: pessoas, organizações, portfólio e mensagens e outras classes auxiliares, objetos e propriedades. Assim é desenhado uma ontologia que pode representar diversos esquemas de branqueamento/lavagem de dinheiro. Em 2010, os autores do artigo [2] apresentaram uma proposta, ontologia e regras, que permitem representar o crime de lavagem de dinheiro, chamada de "minimal model". Com esta representação, os autores pretendem descobrir, através de regras, os diferentes papéis dos intervenientes, e o seu nível de relacionamento (para o caso de uso, é extremamente importante, estabelecer este relacionamento e o seu nível). Adicionalmente, as relações entre empresas também são estabelecidas, por forma a provar relações: entidades, pessoas e acções. No artigo [8] publicado em 2016, os autores desenharam uma ferramenta suportada numa ontologia, que procura representar a informação semântica extraída nas investigações forenses. A ontologia é suportada por três niveis, o nivel "Abstract Knowledge Layer"representa o conhecimento dos peritos, o nível "Knowledge Processing Layer"suporta o conhecimento forense e finalmente o nível "Concrete Knowledge Layer"representa os dados extraidos e armazenados em formato digital.

3 Ontologia PROV-O

A ontologia PROV-O [1] basea-se no modelo de dados PROV-DM [1], usando a OWL2, linguagem que permite definir e instanciar ontologias na Web. Fornecendo um conjunto de classes, propriedades e restrições que podem ser usadas para representar a proveniência das informações originadas por sistemas e contextos diferentes, assim como o seu intercâmbio ao longo de um determinado período de tempo. Dado a sua flexibilidade, podem ser criadas novas classes

e propriedades para diferentes aplicações e domínios. Nos parágrafos seguintes iremos descrever a ontologia, mas recomendamos a leitura do documento de suporte da ontologia, que pode ser consultada em [1] para um conhecimento mais aprofundado. A ontologia foi desenvolvida com base em 3 classes: Entity, Activity e Agent. A classe prov:Entity define um conceito: físico, conceptual, digital ou outro tipo de "coisa", baseado em certas propriedades, podendo estas ser imaginárias ou reais. A classe prov:Activity define algo que ocorre num período de tempo sobre/com as entidades (prov:Entity). Finalmente, a classe prov:Agent é algo que toma a responsabilidade de uma determinada acção sobre determinada actividade (Activity). Na imagem 1pode ser observado as propriedades associadas às 3 classes, onde podemos verificar que algumas, estão relacionadas entre elas e entre si, individualmente. Normalmente, uma actividade está associada a

Figura 1. Entidades, Agentes e Atividades - PROV-O [1]



um período de tempo: Início e Fim. Durante este período de tempo, podemos usar (prov:used) e gerar (prov:wasGeneratedBy) de uma quantidade de entidades (prov:Entity). Por exemplo, a compra de uma determinada propriedade (casa) usa uma determinada quantia de dinheiro e gera um registo de propriedade, associado a um determinado agente (prov:Agent). A classe prov:Activity, possui relações entre si mesma, prov:wasInformedBy, sugerindo que uma actividade pode informar outra actividade assim fornecer dependências entre si. Com esta dependêcia podemos criar uma cadeia de informação baseada apenas nas actividades entre si. O mesmo acontece na classe prov:Entity, com a pro-

priedade prov:wasDerivedFrom, que permite a transformação de uma entidade noutra entidade. Finalmente, o mesmo pode ser aplicado à classe prov:Agent, onde agentes podem ter a responsabilidade por outros agentes e influênciar outros agentes, prov:actedOnBehalfOf, e sobre a responsabilidade sobre actividades (prov:Activity) e entidades(prov:Entity). Claro que a ontologia não se limita a estas classes, contêm classes adicionais e propriedades, fazendo com que seja aplicada em diversos dominios.

4 Investigação Criminal - Branqueamento de capitais

A melhor definição que podemos enunciar para Investigação Criminal, está descrita na Lei de Organização da Investigação Criminal, art.1 da Lei 49/2008 de 27 de Agosto, e define-se pelo "conjunto de diligências que, nos termos da lei processual penal, se destinam a averiguar a existência de um crime, determinar os seus agentes e a sua responsabilidade, descobrir e recolher as provas, no âmbito do processo". No âmbito do acto de branqueamento de capitais existem diversas definições, todas elas têm em comum termos como: encobrimento, dissimulação, sistema económico, origem ilícita de bens. Assim, trata-se de um processo de encobrimento ou dissimulação através de operações, apoiadas no sistema económico/financeiro, por forma a justificar a origem de quantias avultadas de dinheiro provenientes de práticas ilícitas ou criminosas [2]. Existem diversos artigos que explicam o funcionamento do processo de branqueamento de capitais [4] [6] [9], que recomendamos a leitura para uma melhor compreensão de todo o processo e dos seus intervenientes. Basicamente, o branqueamento de capitais baseia-se num processo de ocultação legítima de bens, produtos ou capitais para que no final deste processo estes tenham uma aparência de legalidade. Existe um processo, chamado de "modelo das três fases", aplicado pelo Grupo de Acção Financeira Internacional (FATF/GAFI)[6], composto por:

- **Colocação:** "consiste na introdução dos bens, produtos ou capitais que se pretendem branquear no sistema económico-financeiro, utilizando os mais diversos meios ou instrumentos" [3];
- **Circulação:** "implicará um conjunto de procedimentos que provoquem grande rotatividade de titularidade dos bens, com vista ao maior afastamento possível entre a sua origem e forma de obtenção, e aquele que finalmente ficará na posse dos mesmos." [3];
- **Integração:** "constitui-se com a integração dos bens e/ou dos valores na esfera patrimonial do criminoso a quem os valores são devidos. Completa-se quando os bens ou valores ilícitos surgem com a aparência de lícitos e são usados livremente pelo criminoso, à frente de todos, muitas vezes até com elevada consideração social." [3].

Voltando à definição enunciada anteriormente, a investigação deve identificar três objectivos distintos:

- Identificar a existência de um crime;
- identificar os seus agentes e a suas responsabilidade;
- identificar e recolher provas, estabelecendo a relação entre o acto e o seu autor.

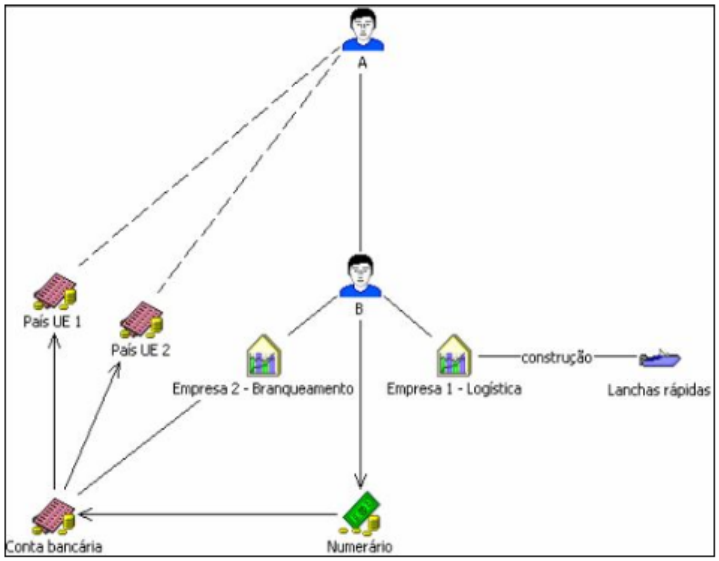
Adicionalmente, o acto de investigação criminal, incluindo o branqueamento de capitais, deve procurar responder às seguintes questões, objectivos de qualquer investigação: quem fez o quê? Onde? Quando? Como? e Porquê?. Adicionalmente, os investigadores procuram padrões que possam levar à detecção de actividade criminosa.

5 A ontologia PROV-O para representação de conhecimento dos crimes de branqueamento de capitais

Estes crimes baseiam-se em entidades fisicas ou não-fisicas (digitais), pessoas interagindo entre elas e com outras entidades, e actividades que geram entidades e interagem com outras actividades. Localizadas no espaço e no tempo. Assim, a ontologia PROV-O que se baseia em classes que representam estas realidades, podendo estabelecer uma rede de conhecimento para representar o crime em questão, note-se que podemos estabelecer uma linha de tempo para todas estas acções, podendo assim estabelecer um nexo de causalidade nas actividades ao longo da linha de tempo. Para que possamos testar a aplicabilidade da ontologia aos crimes de branqueamento de capitais, iremos recorrer a um caso de uso apresentado em [3], que passamos a descrever. "Em Portugal identificaram-se duas empresas com relação no apoio logístico ao tráfico de cocaína e no branqueamento de capitais por parte de uma organização criminosa dedicada ao tráfico internacional de estupefacientes. Uma delas, dedicava-se à construção de evoluídas lanchas rápidas essenciais ao transporte de droga, enquanto que a outra procedia à introdução dos lucros do tráfico de droga no sistema financeiro. Apurou-se que a empresa utilizada para o branqueamento dos capitais, recebia nas suas contas bancárias elevados depósitos em numerário que seguidamente eram disseminados por várias pessoas colectivas e singulares em Portugal e em outros países europeus, escoando deste modo os capitais introduzidos no sistema financeiro. Foi promovido junto da autoridade judiciária a suspensão das contas bancárias das empresas e a apreensão dos valores ali creditados o que originou uma investigação por branqueamento de capitais" [3]. Na figura 2, podemos observar a reprentação do gráfica do caso de uso.

Após a leitura do caso de uso, auxilado pela figura2, podemos enumerar as diferentes; entidades, actividades, e intervenientes. Assim, podemos identificar as classes, da ontologia PROV-O, que podem corresponder a cada uma deles.

Figura 2. Caso de Uso - Branqueamento de capitais [3]



- **prov:Agent:** A e B;
- **prov:Activity:** Numerário, Construção de Lanchas;
- **prov:Entity:** Conta Bancária, Pais UE 1 e 2, Empresa 2 - Branqueamento, Empresa 1 - Logística.

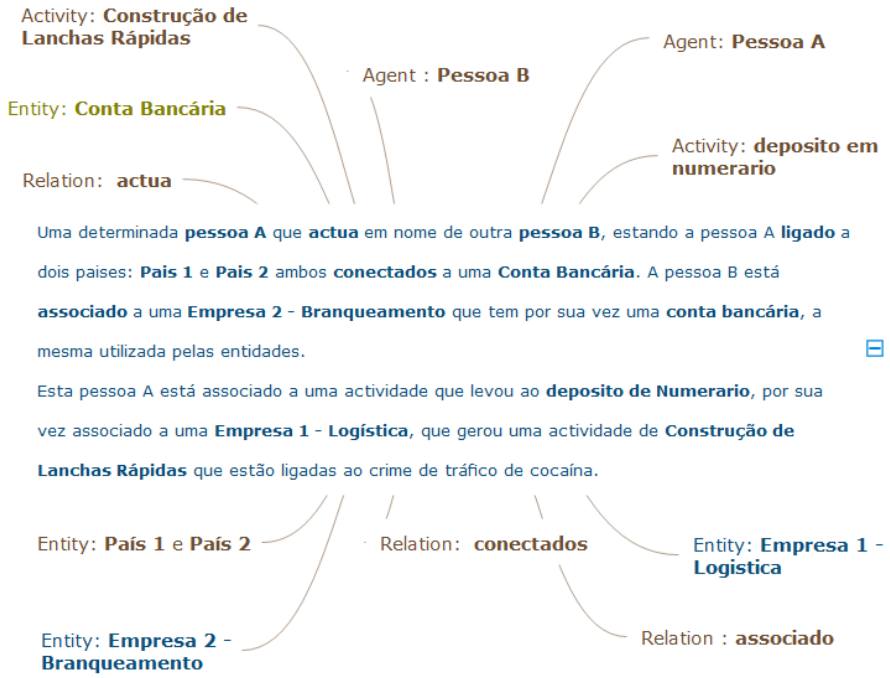
A representação deste caso, está restringida a uma linha de tempo, por isso, a identificação através de datas e tempo é algo que temos de levar em conta, principalmente nas actividades: depósito de dinheiro ou construção de lanchas rápidas. Representação descritiva do caso de uso através da ontologia PROV-O, na figura 4.

Claro que a representação do caso de uso, não se pode limitar apenas ao que demonstramos em cima, pois deve existir regras associadas a este tipo de crime, para que possamos extrair padrões e assim detectar a actividade criminal a partir de um conjunto de operações.

6 Conclusão

Depois da análise realizada à ontologia PROV-O, podemos de alguma forma adaptar ao conhecimento inerente aos crimes de branqueamento de capitais. Dado que a ontologia se baseia em entidades, agentes e actividades, algo que os

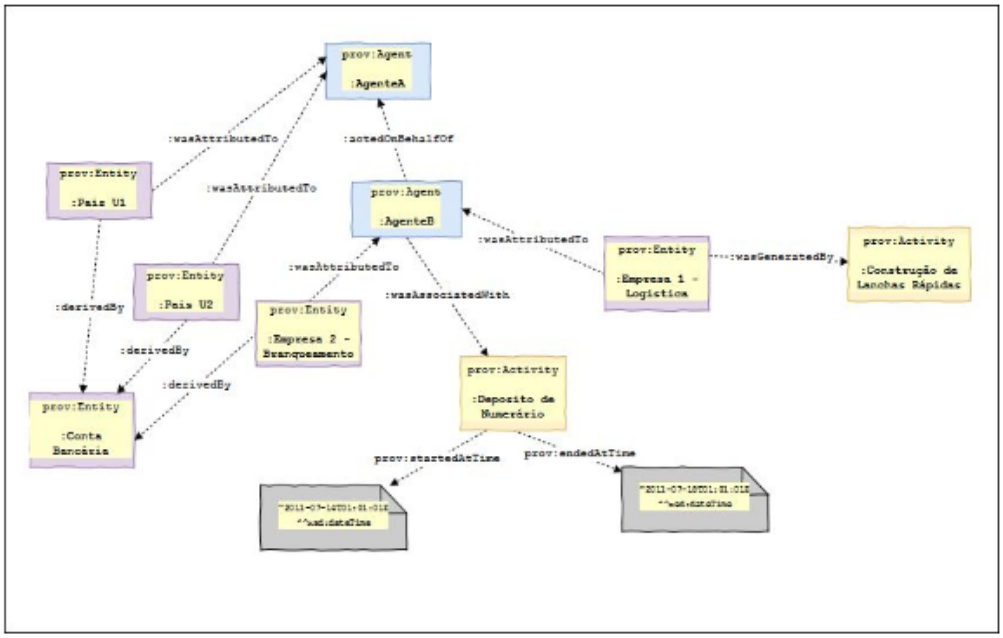
Figura 3. Caso de Uso - Descrição e associação da terminologia da ontologia PROV-O



crimes também se baseiam. Assim como a possibilidade de adicionar propriedades temporais e espaciais, dando assim aos investigador a ferramenta necessária ao desenho de uma linha de tempo. Contudo, e depois de confrontar com estudos anteriores, a ontologia torna-se limitada no que respeita à possibilidade de detecção de esquemas de crime, assim como a falta de algumas classes, objetos e propriedades que possam representar o domínio de uma forma mais exacta. Como trabalho futuro:

- desenvolvimento de regras, por forma a inferir e detectar padrões de esquemas de branqueamento de capitais;
- adicionar mais classes, objetos e propriedades por forma a representar conceitos associados ao domínio, de uma forma mais específica, colocando a ontologia PROV-O num nivel mais abstrato;
- representar a ontologia de uma forma permanente, para assim criar uma base de dados com as instâncias.

Figura 4. Representação do caso de uso da imagem 1 na ontologia [1]



Resumidamente, a ontologia PROV-O tem todas as condições para suportar os conhecimentos abstratos do domínio, dado as classes que a constituem, e a possibilidade de extensão da mesma para uma ontologia de dominio.

Referências

1. Prov-o ontology. https://www.w3.org/TR/2011/WD-prov-o-20111213/, Jan. 2017.
2. J. Bak, C. Jedrzejek, and M. Falkowski. Application of an Ontology-Based and Rule-Based Model to Selected Economic Crimes: Fraudulent Disbursement and Money Laundering. pages 210–224. Springer Berlin Heidelberg, 2010.
3. J. L. Braguês et al. O processo de branqueamento de capitais. *Observatório de Economia e Gestão de Fraude.[Em linha] Porto: Edições Húmus. Disponível em: http://www. fep. up. pt/repec/por/obegef/files/wp002. pdf,[Consult. 25 fev. 2013]*, 2009.
4. A. Chong and F. López-De-Silanes. Money Laundering and its Regulation. 2007.
5. G. Kul and S. Upadhyaya. Towards a Cyber Ontology for Insider Threats in the Financial Sector.
6. D. Masciandaro. Money Laundering: the Economics of Regulation. *European Journal of Law and Economics*, 7(3):225–240, 1999.

7. M. Mehmet and D. Wijesekera. Ontological Constructs to Create Money Laundering Schemes.
8. R. Merkel, C. Krätzer, M. Hildebrandt, S. Kiltz, S. Kuhlmann, and J. Dittmann. A Semantic Framework for a better Understanding, Inves- tigation and Preventionof Organized Financial Crime. *Lecture Notes in Informatics*, 2016.
9. Peter J. Quirk. Money Laundering: Muddying the Macro economy. *Finance & Development*, 1997.
10. Q. Rajput, N. S. Khan, A. Larik, and S. Haider. Ontology Based Expert-System for Suspicious Transactions Detection. *Computer and Information Science*, 7(1), 2014.

Ontology-Based Framework Applied to Money Laundering Investigations

Gonçalo Carnaz[1], Vitor Nogueira[1] and Mário Antunes[2,3]

[1] Informatics Department, University of Évora
d34707@alunos.uevora.pt, vbn@di.uevora.pt
[2] School of Technology and Management, Polytechnic Institute of Leiria
mario.antunes@ipleiria.pt
[3] Center for Research in Advanced Computing Systems (CRACS), INESC-TEC;
University of Porto mantunes@dcc.fc.up.pt

Abstract. Criminal investigations face a deluge of structured and unstructured data obtained from heterogeneous sources like forensic reports or wiretap transcriptions. In these cases, finding relevant information can be a complex task. Ontologies have been successfully applied to several domains including legal, cybercrime and digital forensics. In this paper[4] it is proposed a framework based on ontology engineering, that provides an unified approach to represent and reason with the criminal investigation data. Moreover, this framework is applied to the specific use case of money laundering.

Keywords: ontology, knowledge representation, criminal investigation

1 Introduction and Motivation

Over the years, with the massive introduction of Information and Communications Technology (ICT) in different professional areas, where users leave their digital fingerprint, bringing new challenges to computer scientists. Crime, however reprehensible, is an activity that uses ICT as a tool of crime, or as proof of it. Criminal polices in general are currently facing new challenges, namely the huge amount of data produced during their investigations, resulting from heterogeneous sources. Concealment, hiding, dissimulation, economic system, illicit origin of assets are terms commonly used in the context of money laundering related crimes. These crimes can also be associated with other crimes, such as drug trafficking.

Computer science has a wide set of tools that may be used to automate analysis and correlation of investigations documents. Some of those tools include frameworks to retrieve data and to represent knowledge, as well as to collect

[4] This paper is for the assessment of Doctoral Seminar 1.

evidences and provides decision making during investigations.

Data is acquired daily as the occurrences and evidences take place. Depending on the type of crime, the types of sources can be challenged, regarding the origin of the documents, like paper, digital reports, handwritten transcripts of interrogations, social networks transcript messages and forensic logs, just to mention a few.

The contribution of this paper is thus to answer the following research question: *How to design and represent the information inherent to documents collected in money laundering investigations?*

Some questions, arises from research question above:

- It is possible to design a knowledge base, based on an ontology that represents the knowledge inherent to money laundering crimes in Portuguese Legal System;
- how can we detect patterns in the knowledge base that lead to money laundering schema's;
- how to find relevant data;
- how to deal with evidences sources that may support knowledge base;
- which visualization format will support users questions.

The remaining sections of this paper are organized as follows: section 2 depicts money laundering crimes and stages; different approaches to digital evidences analysis are described in section 3; in section 4.2 we describe approaches related with ontologies. In section 5 we present a framework to deal with money laundering. Finally, in section 6, we discuss the conclusions and delineate the future work.

2 Money Laundering Crimes

The best definition that can be enumerated for Criminal Investigation is described in the Criminal Investigation Organization Law, art. 1 of Portuguese Law 49/2008 of August 27 [13], and is defined by the *"set of measures that, under the terms of criminal procedural law , are intended to inquire the crime existence, to determine its agents and its responsibility, to discover and collect evidence in the course of the proceedings..."* [13].

In money laundering there are several definitions, all of which have in common the following main terms: concealment, dissimulation, economic system, illicit origin of assets. Thus, it is a process of cover-up or dissimulation through operations, supported by the economic/financial system, as a result of the large amount of money arising from illicit or criminal practices [7]. Basically, money laundering is based on a process of legitimate concealment of goods, products or

capital so that, at the end of the process, they have the appearance of legibility. Money laundering is supported by a process, called the "three-step model", implemented by the Financial Action Task Force (FATF) [36], described next:

- **Placing:** *"consists of the introduction of goods, products or capital that are to be laundered into the economic-financial system, using the most diverse means or instruments"* [7];
- **Circulation:** *"it will imply a set of procedures that provoke great rotation of ownership of the goods, with a view to the widest possible distance between their origin and the way of obtaining them, and the one that will eventually remain in their possession."* [7];
- **Integration:** *"is constituted by the integration of assets and/or values in the patrimonial sphere of the criminal to whom the values are due. It is completed when the illicit goods or values appear with the appearance of licit and are used freely by the criminal, Ahead of everyone, often even with high social consideration."* [7]

Returning to the definition stated above, we must identify three distinct objectives:

- the existence of a crime;
- their agents and their responsibilities;
- collect evidence, establishing the relationship between the act and its author.

Furthermore, any criminal investigation, including money laundering, should answer the following questions, the purpose of any investigation: Who?, Where?, When?, How? or Why?. In addition, researchers should look for patterns that may lead to the detection of criminal activity.

3 Digital Evidences Frameworks

The literature on frameworks for digital evidences analysis shows a variety of approaches, from academic to industrial ones. The following paragraphs will give us a selected work discussion focus in the academic approach. In [45] POLESTAR is described as a framework for knowledge management and collaboration tool for analysts, providing a framework from text documents and creating a documents repository to analysis. One of the main framework features is anomaly detection, alerting experts if any anomaly is detected in data retrieved from text documents. In [46], authors define an architecture that abstracts digital evidence, retrieving those evidences from multiple sources. They also realize that past reconstruction would be a requirement for the system, which facilitates the investigators' theories. The authors in [41] define a framework that crawls into Web blogs looking for relevant information. A three layer infrastructure is defined to support crawling and to store information for further analysis.

In [1] authors propose central repositories of integrated data from one or more heterogeneous sources concepts into a framework supported by a 5-steps interactive process: 1) Data identification; 2) Business Data Model Design; 3) Data Warehouse Model Design; 4) Testing and Analysing and 5) Data Marts Models Design. Supported by these 5 steps, they designed a relational tool that analyses crime activity, also organized police reports into a data warehouse, named by "police logs".

In [27] authors focuses their work in social criminal networks understanding, designing a framework that fetches data retrieved from Web and documents, as a result, they design criminal networks, detect crime hot stops and profiling criminal steps.

The authors [32] analyse criminal networks based on communication logs, they have used a interactive process for criminal network construction from smartphone call logs, based into phases: First one, data are clean by expert officers and data engineers; second, added metrics by social networks experts and finally a analysis is performed over the network supported by machine learning algorithms. This method is supported, initially, by human intervention and for learning, algorithms are applied to retrieve knowledge. A different approach made by [18], focus in criminal network visualization supported by mobile calls logs reconstruction. Every day, police officers produce text documents related to crime investigations and victims reports, the paper [51] authors developed a visual analytical tool that identifies entities on those documents and visualise them in multiple views (coordinated).

In [9] a system to predict survival techniques after a terrorist attack is described, by crawling into twitter, analysing the propagation of re-tweeting to map the necessity of survival, as a mechanism of defense. In [38], based on use scenario of Point-of-Sale (POS) Skimming, authors proposed a semantic framework to structured knowledge related to financial crimes.

Finally, there are industrial approaches, such as Analyst Notebook[5], Xanalysis Link Explorer [6] and Palantir [7] that may be consider to review.

4 Overview of Ontologies

Historically, the term "ontology" has its roots in two Greek words: "ontos", being, and "logos", word. Being the original word "category", applied by Aristotle in the sense of classification. Aristotle developed a list of categories that served as the basis for classifying any entity, dividing reality into entities: (i) individual substances and (ii) their qualities.

[5] http://www-03.ibm.com/software/products/pt/analysts-notebook
[6] http://www.xanalys.com/products/link-explorer/link-explorer-analysis/
[7] https://www.palantir.com/

From a philosophical point of view, the Oxford Dictionary of Philosophy, ontology is defined as: *"[...] the term derived from the Greek word for 'being', most used since the seventeenth century to refer to Branch of metaphysics that concerns what exists"*. Gruber defined: *"An ontology is an explicit specification of a conceptualization"* [24]. In this ontology, definitions associate names of entities in the universe of discourse (eg, classes, relations, functions, etc. with texts that describe what the names mean and the formal axioms that restrict the interpretation and use of those terms) [2]. From Gruber's definition, the term conceptualization as emerged, which corresponds to objects, concepts properties and other entities, that can be represented in several domains of knowledge. Therefore, conceptualization can be interpreted as abstraction, to represent the world in a simplified way. In 1997, Borst [6], defines ontology as: *"[...] Ontologies are defined as the formal specification of a shared conceptualization"* [6], while Gruber [24] defines ontology as: *"[...] An ontology is an explicit specification of a conceptualization [...] "* [24]. In computer science, ontologies have been developed in artificial intelligence in order to facilitate the sharing and reuse of information.Therefore, ontologies are applied to a wide range of computer science applications, and a significant contribution to the representation of the concepts, relationships and properties associated with the knowledge acquired in the different domains, so there are different areas of ontology using it, from Knowledge management [15] to the medical area [50].

4.1 Cybercrime and Forensic Ontologies

This section is an overview of related works that focus the cybercrime and forensic ontologies. In [54] proposed a dynamic and real-time forensic model based on ontologies and context information, where model is based on the authentication method that supports user's authentication, depending on the context. Therefore, authors added an ontology that describes the entities, authorizations and rules involved. Any police investigation process needs proofs, in [44] DCoDeOn ontology is defined, that allows preserving the cybercrime evidence, the so-called Chain of Custody[8], the authors defined a taxonomy diagram[9] that allows Chain of Custody representation. In [52] developed an ontology that seeks to represent knowledge in computer forensics field, namely:

- Digital forensic domain representation;
- Disciplines: software forensics, network, computer, database, multimedia and devices;

[8] in legal and police contexts, refers to the chronological documentation, showing analysis and disposition of physical or electronic evidence.

[9] a classification and naming in a ordered system that indicates relationships, in a form of a diagram.

- Sub-disciplines, such as: operating systems and applications forensics, or mobile forensics;
- Objects, such as forensic objects, i.e. web-services or authentication services;
- Sub-objects, i.e. access control systems with their logs.

In [43] authors proposed an ontology to supports safe operations in cyberspace. In this work, besides demonstrating the concepts related to the domain, they added the human factor as an important part of this technological field. They created the CRATELO ontology:

- Top level: the DOLCE SPRAY ontology allows the natural language understanding, capturing the primitive concepts inherent to the language;
- Middle level: the SECCO ontology defines security concepts in cyberspace;
- Lowest level: the OSCO ontology represents operations in cyberspace.

In [33] authors describe an approach to solve one of the main objectives of computer forensics: cause-effect in the acquisition of digital evidence.Therefore, authors developed a platform based on an ontology consisting in two layers: hardware and software.

- The hardware sub-layer: representing the digital equipment used in the investigations;
- The software sub-layer: representing forensic analysis tools and operating systems.

The authors [22] created an ontology that represents the culture around cyberspace security, and the relations between different entities. The knowledge base of this ontology has resulted in the information acquired about the culture on cyberspace, based on campaigns in the communities (users). The use of social networks is not limited to recreational or professional purposes, but also to criminal activities. Therefore, platforms like Facebook, YouTube, Linkedin, etc. are used by criminal investigation entities as data sources for analysis (future or even in real time), for the detection and proof of crime. The ontology SC-Ont [29] was proposed to support the criminal domain and its relations, based on social networks. The smartphone, one of the most used devices for communication. Therefore, the F-DOS ontology [34] allows the abstraction between the user and the data collected by smartphones. This ontology consists of:

- A core ontology, where the essential concepts of the domain are presented;
- Other domain ontologies: contacts, messages and research.

There is a growing concern to represent, analyze and process data collected in criminal investigations. Authors [12] proposed an ontology that seeks to answer essential questions in the presentation of evidence in Court House: what, who, when, where, why and how.

The WikiCrimes platform [20] [21] allows the collaborative use based on maps manipulation, in order to register the criminal movements. The architecture is based on two ontologies: Crime and Reputation. In [44] designed an ontology applied to criminal investigation in cyberspace, with the categories of cybercrime, laws, evidence and information of suspects. In these domains there is the difficulty to relate the type of crimes and the collection of evidence, with this ontology, the authors try to represent this correlation and thus to detect the associated crimes and the evidences evidenced. Based on a Semantic Web framework, [16] presented the problems inherent in the integration and correlation of digital evidence, trying to present the steps necessary for the representation, aggregation and integration of this digital evidence in an ontology. In [39] described an event-based ontology for cybercrime, defined the events and their relationships using 6-tuples[10]: Action, Participant, Time, Location, Instrument and Good. He divided the relationships between classified and non-classified. Starting from an example of online banking fraud, they proposed the OBM ontology [11] to map out criminal organizations and identify malware developers. Finally, they also defined rules of inference based on empirical knowledge that would meet some of the needs of the forensic analyst.
From the wide spectrum of ontologies describe below, from cybercrime to forensics domains, can helps us implementing ontologies regarding forensics evidences knowledge representation.

4.2 Legal Ontologies

This section is an overview of related works that focus the legal ontologies. In [3] proposed an ontology as a support for the representation of crime and/or criminal activity in Italy, aims to be an attempt to solve some problems found in ongoing projects that were not based on ontologies and that did not have a conceptual definition of a knowledge base in order to achieve a conceptual framework for the various projects, added a domain knowledge also draw the classes that allow the ontological representation of the concept of crime, they defined a suspect/criminal - a person who acts in a manner punished by criminal law, with a given behavior in a given time interval - Event, and the penalty applied to the perpetrated act. It will thus support the management of documents as metadata, identify and suggest a crime hypothesis to the Judge, and semantically map criminal laws using the XML[11] language. Authors [48] developed the integration of different ontologies, different domains, representing the heterogeneous data gathered in the different information and communication technologies, in order to solve the lack of specialization of some researchers in the domains in

[10] a ordered list of elements
[11] eXtensible Markup Language

question, leading to the creation of the FORE ontology. On the other hand, in [8]proposed two basic ontologies applied to the legal domain:

- FOLaw is based on legal knowledge and seeks to represent this same knowledge;
- LRI - CORE supports the construction of structured legal domains, to allow automatic indexing of legal texts.

LKIF ontology [26] emerged as part of an architecture for information systems in the legal domain. The ontology has two requirements:

- Translation between legal knowledge base represented in different formats and formalisms;
- Formal representation as part of an information system architecture.

This ontologies are represented diverse legal / juridical knowledge, such as: documents, norms, laws. Another important requirement was the attention to the different levels of knowledge of the users. From papers related below, a sorted of ontologies were developed to support knowledge from the legal domain, that we will take in account for our framework, in case we need. Thus, legal domain differ from one country to another, we have to adapt our ontology to country legal system.

Money Laundering In [30], authors define the steps for an ontology, named FF POIROT, which represents knowledge in the field of financial crime. In [47] tries to represent a suspicious financial transactions through an expert system, based on an ontology and a set of rules. Following the rules of design applied to ontologies, the authors created a set of classes, objects and properties that represent the transactions to be processed by the expert system. Additionally, a set of rules, using SWRL (Semantic Web Rules Language), in order to infer new knowledge through existing knowledge. In [37], the authors define an ontology that can map the knowledge inherent money laundering , constructing an ontology that can help discover money laundering schemes. They are defined entities: people, organizations, portfolio and messages and other auxiliary classes, objects and properties. In [4] presented a proposal, ontology and rules, that allow to represent the crime of money laundering, called "minimal model". With this representation, the authors intend to discover, through rules, the different roles of the actors, and their level of relationship (for the use case, it is extremely important, establish this relationship and its level). In addition, relationships between companies are also established, in order to prove relationships: entities, people and actions. In [38] the authors designed a tool supported by an ontology, that tries to represent the semantic information extracted in the forensic investigations. The ontology is supported by three levels: "Abstract Knowledge

Layer" represents the knowledge of the experts; "Knowledge Processing Layer" supports forensic knowledge;"Concrete Knowledge Layer" represents the data extracted and stored in digital format.

5 Ontology-Based Framework for Money Laundering

The methodology proposed is to transform the informal and unstructured data retrieved in heterogeneous police data sources, such as police reports, into a structured knowledge. In the proposed framework we integrate the different kind of data sources, and with that, we can correlate data to help police investigations, like detecting different stages of money laundering done by different entities, for example: smurfing schema [35], all this supported by a defined ontology that will represent all knowledge associated to domain. The framework is represented in Figure 1. It is important to mention here that the framework can collect data from data-sources in Portuguese language.

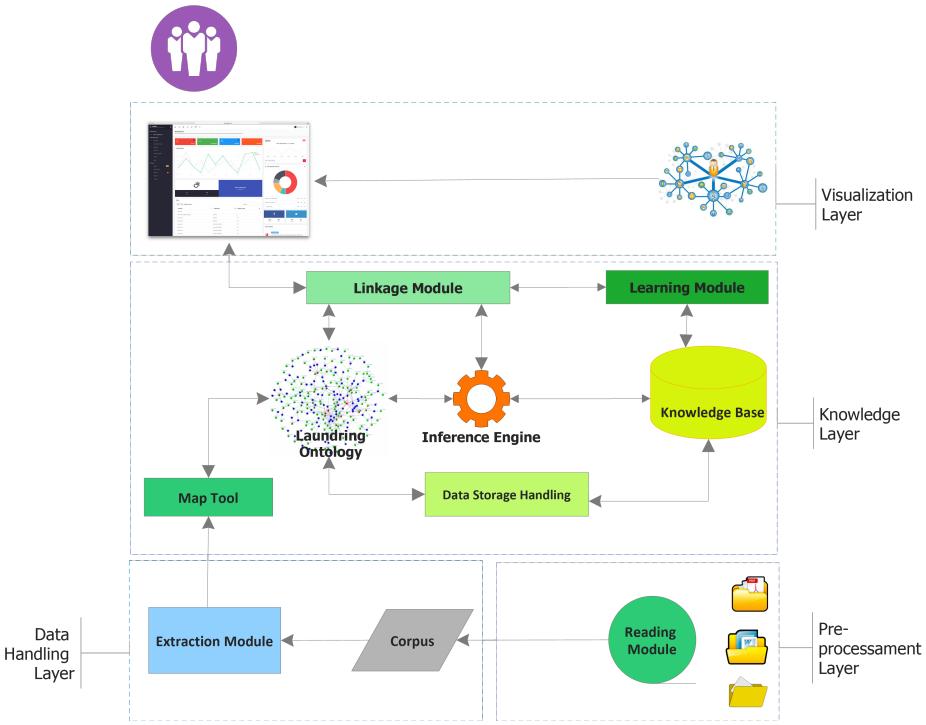


Fig. 1. Ontology-based framework for money laundering schema

5.1 Pre-processing Layer

Police repositories store millions of reports pages on crimes, offenders, and other intelligence. Thus, Pre-Processing layer define the data sources to collect, analyze and process all unstructured data. Currently, all police departments dedicated to money laundering produce reports, with different data types: text or numbers, and different formats: spreadsheets, text documents or forensic logs. There are some challenges associated with this layer:

- Deal with different data sources formats and types;
- How to deal with the asynchronous feeding, because all police cases are continuous updating evidences;

Also, this is a big data issue, framework must deal with the four big data dimensions: volume, variety, velocity and veracity [28]. To solve the enumerated challenges, a reading module will be added to framework, that will systematically browses documents, to retrieve and cleaning data. This module fits in batch processing definition, because documents represent chunks [28], that can be processed in parallel. Police reports are retrieved into plain text, forming the text corpus. This process is necessary for cleaning text purposes to exclude any noise from that, such as images or videos.

5.2 Data Handling Layer

This layer supports corpus analysis that was created from previous layer, there is a main challenge associated with this layer:

- Extract entities and relations in Portuguese language sources;

This is done by the Extraction Module. Using natural language processing (NLP) to retrieve information, such as entities and relations from text corpus. Done in two phases:

- Pre processing phase: remove unnecessary data, in order to organize the information extracted so that classification will be simpler and more efficient. For this, we use NLP techniques, such as tokenization, lower case, stopwords removal or stemming [23].
- Feature and classification phase: named entity recognition [40] techniques will be performed to classify entities and relations.

There are some studies [14] [49] [10] [31] to support natural language process understanding, in English, and also in Portuguese [42] [19]. Therefore, this layer will retrieve and identify entities and relations, analysing each sentence trying to extract context meaning on each word, in Portuguese language.

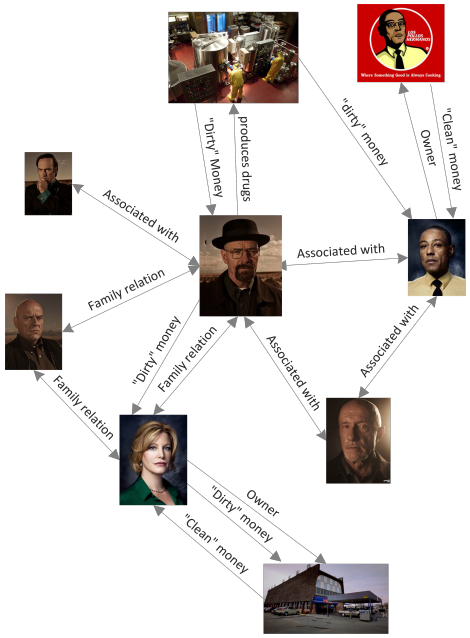


Fig. 2. Use Case - Breaking Bad

5.3 Knowledge Layer

The knowledge layer main objective is to define an ontology that allows the knowledge representation. In order to size an ontology that may support the knowledge inherent to the crime of money laundering, a use case was taken from the well-known American TV series - Breaking Bad [12]. In the Fig 2: **'Walter White'** that is associated with an **'Organization'**, maintaining family relationships, with **'brother-in-law'** and **'wife'**, delivers **'sums of money'** to the **'wife'**, who makes them go through the **'car wash'**, that she is the **'owner'**, thus clearing the money, which comes from **'drug traffic'**. Associated with, there are external signs of wealth, such as the purchase of a **'high-powered car'**, by **'Walter White'**. Therefore, from observation of use case, we need to dimension a ontology that supports knowledge inherent to the domain, representing actors, objects, actions, time and other relevant information. There are some features, that we can implement or improve from previous works:

[12] http://www.imdb.com/title/tt0903747/

- LKIF [26] ontology is an example applied to legal domains, representing legal/juridical knowledge, that can be suitable to our work, with a thesaurus[13] enriching our ontology;
- how time is represented [3];
- answer essential questions in the presentation of evidence in Court House [20].

In order to transform the collected data, commonly expressed is heterogeneous data formats, into their semantic representation, and match the extracted data to the instances, a Map Tool is defined to perform this task. Therefore, the ontology must map all entities and relations, also should allow the instantiation all data into ontology schema.

The ontology must be based on entities, agents and activities, something that crimes are also based on, as we can see in use case above. As well as the possibility of adding temporal and spatial properties, giving the possibility to draw a events timeline [17]. Summing up, ontology must reflect domain terminologies, entities, events, actors and relations between each others, with event time related. A inference engine will be added to define rules and perform inference queries against defined ontology, there are some related studies [25] [53] that may support our implementation. A data storage handling will be used to manage the CRUD[14] operations that instantiate data within the semantic representation model to a permanent database, with the semantic model as data schema, this database must reflect the heterogeneous environment.

The linkage module will support connections with different modules, acting as a bridge, between: visualization and knowledge layers, with laundering ontology, inference engine and the learning module.

Finally, learning module will support machine learning algorithms to learn from previous data and give a substantiated hypotheses to police activities, like a recommendation system, related to money laundering.

5.4 Visualization layer

This layer aims to knowledge visualization, how user will interact with extracted knowledge and visualises graph links and patterns. One way of content visualization is displaying it as a graph, because highlights patterns, and show clusters and connections, tools like [5]. Therefore, this block must give a visual analysis tool to:

- understand data that we are collecting ;
- understand relations between data elements;
- perform queries to knowledge base and visualize them;
- visualize networks of crime, based on data relations created in layers below.

[13] "lists words grouped together according to similarity of meaning" in Wikipedia - https://en.wikipedia.org/wiki/Thesaurus

[14] Create, Read, Update, Delete

5.5 Tools

The deployment of the framework benefits from using a wide set of tools in the various topics involved. The following list describes theses tools:

- Tika (`https://tika.apache.org/`) - detect and extract metadata and text from different sources.
- GATE (`https://gate.ac.uk/`) - retrieve data from text corpus using NLP.
- Protege (`protege.stanford.edu/`) - create, map and management of ontologies.
- CounchDB (`couchdb.apache.org/`) and Neo4j (`https://neo4j.com`) - database creation and management.
- Jena (`https://jena.apache.org`) - to engine inference.
- Gephi (`https://gephi.org/`) - for graph visualization.

6 Conclusion

In this paper we have provided an overview of different approaches for knowledge representation using ontologies. We made a comprehensive study of the literature and identified several challenges regarding the use of ontologies applied to the development of frameworks for crime investigation. That is, the amount of data related with criminal investigation, coming from distinct sources, are challenging computer science research to deploy and develop ontology-frameworks to identify and correlate terms and subjects related with a specific kind of crimes: money laundry.

We have proposed a framework based on a ontology to support knowledge representation related to money laundering. Since Portuguese is the default language that brought us new challenges regarding lexical and semantic features. The framework is composed by several components, organized in four layers: Pre-processing, data handling, knowledge and visualization. Based on previous research and domain requirements related to money laundering analysis, we present our initial design for the framework, from evidences retrieval, crossing knowledge representation and visualisation.

Future work consists on developing this framework and test it with real world scenarios in money laundering.

References

1. F. Albertetti and K. Stoffel. From Police Reports to Datamarts: Towards a Crime Analysis Framework. In *Proceedings of the 5th International Workshop, IWCF 2012, Tsukuba, Japan*, pages 48–59, 2012.

2. M. B. Almeida and M. P. Bax. Uma visão geral sobre ontologias: pesquisa sobre definições, tipos, aplicações, métodos de avaliação e de construção. *Ciência da Informação*, 32(3):7–20, 2003.
3. C. Asaro, M. A. Biasiotti, P. Guidotti, M. Papini, M.-T. Sagri, D. Tiscornia, and L. Court. A Domain Ontology: Italian Crime Ontology. *Proceedings of the ICAIL 2003 Workshop on Legal Ontologies and Web based legal information management*, pages 1–7, 2003.
4. J. Bak, C. Jedrzejek, and M. Falkowski. Application of an ontology-based and rule-based model to selected economic crimes: fraudulent disbursement and money laundering. In *International Workshop on Rules and Rule Markup Languages for the Semantic Web*, pages 210–224. Springer, 2010.
5. M. Bastian, S. Heymann, M. Jacomy, et al. Gephi: an open source software for exploring and manipulating networks. *ICWSM*, 8:361–362, 2009.
6. W. N. Borst. *Construction of Engineering Ontologies for Knowledge Sharing and Reuse*, volume PhD. 1997.
7. J. L. Braguês et al. O processo de branqueamento de capitais. *Observatório de Economia e Gestão de Fraude.[Em linha] Porto: Edições Húmus. Disponível em: http://www. fep. up. pt/repec/por/obegef/files/wp002. pdf,[Consult. 25 fev. 2013]*, 2009.
8. J. Breuker and R. Hoekstra. Epistemology and ontology in core ontologies: FOLaw and LRI-Core, two core ontologies for law. *Proceedings of the EKAW04 Workshop on Core Ontologies in Ontology Engineering*, pages 15–27, 2004.
9. P. Burnap, M. L. Williams, L. Sloan, O. Rana, W. Housley, A. Edwards, V. Knight, R. Procter, and A. Voss. Tweeting the terror: modelling the social media reaction to the Woolwich terrorist attack. *Social Network Analysis and Mining*, 4(1):1–14, 2014.
10. E. Cambria and B. White. Jumping nlp curves: a review of natural language processing research [review article]. *IEEE Computational Intelligence Magazine*, 9(2):48–57, 2014.
11. R. Carvalho, M. Goldsmith, and S. Creese. Applying semantic technologies to fight online banking fraud. In *Intelligence and Security Informatics Conference (EISIC), 2015 European*, pages 61–68. IEEE, 2015.
12. J. Ćosić and Z. Ćosić. The Necessity of Developing a Digital Evidence Ontology. *23th Central European Conference on Information ...*, (January 2012):325–330, 2012.
13. D. da República. Lei n.º 49/2008 de 27 de Agosto - Lei de Organização da Investigação Criminal, 2008.
14. S. Decherchi, S. Tacconi, J. Redi, A. Leoncini, F. Sangiacomo, and R. Zunino. Text clustering for digital forensics analysis. In *Computational Intelligence in Security for Information Systems*, pages 29–36. Springer, 2009.
15. J. Domingue. Tadzebao and Webonto: Discussing, Browsing and Editing Ontologies on the Web. In *11th Knowledge Acquisition Workshop*, page 20, 1998.
16. S. Dosis, I. Homem, and O. Popov. Semantic representation and integration of digital evidence. *Procedia Computer Science*, 22:1266–1275, 2013.
17. V. Ermolayev, S. Batsakis, N. Keberle, O. Tatarintseva, and G. Antoniou. Ontologies of time: review and trends. *Int. J. Comput. Sci. Appl*, 11(3):57–115, 2014.

18. E. Ferrara, P. De Meo, S. Catanese, and G. Fiumara. Visualizing criminal networks reconstructed from mobile phone records. *CEUR Workshop Proceedings*, 1210, 2014.
19. A. M. G. Ferreira. Ontospares: da linguagem natural às ontologias. contributos para a classificação automática de dados históricos (séc. xvi-xviii). 2016.
20. V. Furtado, L. Ayres, M. de Oliveira, E. Vasconcelos, C. Caminha, J. D'Orleans, and M. Belchior. Collective intelligence in law enforcement - The WikiCrimes system. *Information Sciences*, 180(1):4–17, 2010.
21. V. Furtado, L. Ayres, M. D. Oliveira, C. Gustavo, and J. Oliveira. Towards Semantic WikiCrimes: Motivation and Goals. *AAAI Spring Symposium: Social Semantic Web: Where Web 2.0 Meets Web 3.0*, pages 27–32, 2009.
22. N. Gcaza, R. V. Solms, and J. V. Vuuren. An Ontology for a National Cyber-Security Culture Environment. *Proceedings of the Ninth International Symposium on Human Aspects of Information Security & Assurance (HAISA 2015)*, (Haisa):1–10, 2015.
23. C. A. Gonçalves, C. T. Gonçalves, R. Camacho, and E. C. Oliveira. The impact of pre-processing on the classification of medline documents. In *PRIS*, pages 53–61, 2010.
24. T. R. Gruber. A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5(2):199–220, 1993.
25. V. Haarslev and R. Möller. Racer: A core inference engine for the semantic web. In *EON*, volume 87, 2003.
26. R. Hoekstra, J. Breuker, M. Di Bello, and A. Boer. The LKIF core ontology of basic legal concepts. *CEUR Workshop Proceedings*, 321:43–63, 2007.
27. J. Hosseinkhani, S. Chaprut, and H. Taherdoost. Criminal network mining by web structure and content mining. *Advances in Remote Sensing, Finite Differences and Information Security. In Proceedings of the 11th WSEAS International Conference on Information Security and Privacy (ISP '12)*, pages 210–215, 2012.
28. H. Hu, Y. Wen, T. S. Chua, and X. Li. Toward scalable systems for big data analytics: A technology tutorial. *IEEE Access*, 2:652–687, 2014.
29. E. Kalemi and S. Yildirim-Yayilgan. Ontologies for Social Media Digital Evidence. *International Journal of Computer, Electrical, Automation, Control and Information Engineering*, 10(2):335 – 340, 2016.
30. G. Kul and S. Upadhyaya. Towards a cyber ontology for insider threats in the financial sector. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications*, 6(4):64–85, 2015.
31. W. G. Lehnert and M. H. Ringle. *Strategies for natural language processing*. Psychology Press, 2014.
32. X. Lou. Criminal Network Analysis with Interactive Strategies : A Proof of Concept Study using Mobile Call Logs.
33. A. Luthfi. The Use of Ontology Framework for Automation Digital Forensics Investigation. *International Journal of Computer, Control, Quantum and Information Engineering*, 8(3):423–425, 2014.
34. N. M. Karie. Building Ontologies for Digital Forensic Terminologies. *International Journal of Cyber-Security and Digital Forensics*, 5(2):75–82, 2016.

35. J. Madinger. *Money laundering: A guide for criminal investigators.* CRC Press, 2011.
36. D. Masciandaro. Money Laundering: the Economics of Regulation. *European Journal of Law and Economics*, 7(3):225–240, 1999.
37. M. Mehmet and D. Wijesekera. Ontological constructs to create money laundering schemes. In *CEUR Workshop Proceedings*, volume 713, 2010.
38. R. Merkel, C. Kraetzer, M. Hildebrandt, S. Kiltz, S. Kuhlmann, and J. Dittmann. A semantic framework for a better understanding, investigation and prevention of organized financial crime. In *Sicherheit*, pages 55–66, 2016.
39. M. Mudholkar. A Study on Significance of Event Ontology Approach in Web Crime Mining. 2(2):298–306, 2013.
40. D. Nadeau and S. Sekine. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30(1):3–26, 2007.
41. M. Naghavi. A Proposed Architecture for Continuous Web Monitoring Through Online Crawling of Blogs. *International Journal of UbiComp*, 3(1):11–20, 2012.
42. G. Neto and A. Ferraz. Sentimentalista: Um framework para análise de sentimentos baseado em processamento de linguagem natural. 2016.
43. A. Oltramari, L. F. Cranor, R. J. Walls, and P. McDaniel. Building an ontology of cyber security. *CEUR Workshop Proceedings*, 1304:54–61, 2014.
44. H. Park, S. Cho, and H.-C. Kwon. Cyber Forensics Ontology for Cyber Criminal Investigation. In M. Sorell, editor, *Forensics in Telecommunications, Information and Multimedia: Second International Conference, e-Forensics 2009, Adelaide, Australia, January 19-21, 2009, Revised Selected Papers*, pages 160–165. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009.
45. N. J. Pioch and J. O. Everett. POLESTAR: Collaborative Knowledge Management and Sensemaking Tools for Intelligence Analysts. *Proceedings of the 15th ACM International Conference on Information and Knowledge Management*, pages 513–521, 2006.
46. S. Raghavan, A. Clark, and G. Mohay. FIA: An open forensic integration architecture for composing digital evidence. *Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering*, 8 LNICST:83–94, 2009.
47. Q. Rajput, N. S. Khan, A. Larik, and S. Haider. Ontology based expert-system for suspicious transactions detection. *Computer and Information Science*, 7(1):103, 2014.
48. B. Schatz, G. Mohay, and A. Clark. Generalising Event Forensics Across Multiple Domains. *Australian Computer Network and Information Forensics Conference*, pages 1–9, 2004.
49. R. Sharnagat. Named entity recognition: A literature survey. *Center For Indian Language Technology*, 2014.
50. K. Sowkarthikaa and V. P. Sumathi. A Survey of Ontologies on Disease Classification. *International Journal of Science and Research (IJSR)*, 5(4), 2016.
51. J. Stasko, C. Görg, Z. Liu, and K. Singhal. Jigsaw: Supporting investigative analysis through interactive visualization. *VAST IEEE Symposium on Visual Analytics Science and Technology 2007, Proceedings*, (March):131–138, 2007.

52. A. M. Talib and F. O. Alomary. Towards a comprehensive ontology based-investigation for digital forensics cybercrime. *International Journal on Communications*, 5(5):263–268, 2015.
53. Z. Wu, G. Eadon, S. Das, E. I. Chong, V. Kolovski, M. Annamalai, and J. Srinivasan. Implementing an inference engine for rdfs/owl constructs and user-defined rules in oracle. In *Data Engineering, 2008. ICDE 2008. IEEE 24th International Conference on*, pages 1239–1248. IEEE, 2008.
54. W. Yang. Dynamic Forensics Model based on Ontology and Context Information. 10(2):270–272, 2013.

Enlisting GPU Power for Constraint Solving

Pedro Roque, Vasco Pedro, and Salvador Abreu

Universidade de Évora/LISP
d11735@alunos.uevora.pt, vp@di.uevora.pt, spa@di.uevora.pt

Abstract. Applying parallelism to constraint solving seems a promising approach and it has been done with varying degrees of success. Early attempts to parallelize constraint propagation, which constitutes the core of traditional interleaved propagation and search constraint solving, were hindered by its essentially sequential nature. Recently, parallelization efforts have focussed mainly on the search part of constraint solving, as well as on local-search based solving.
The most obvious source of parallelism are multicore processors and shared-memory multiprocessors, which impose the least burden on the developer. Lately, another source of parallelism has become pervasive, in the guise of GPUs, able to run thousands of parallel threads, and they have naturally drawn the attention of researchers in parallel constraint solving. And even if it turned out that the computational model of a GPU may be ill-suited to backtracking search, and that its parallelism potential may not be fully exploitable, they can still be a valuable resource.
In this paper, we present ongoing work on a parallel solver, able to harness some of the GPUs' computational power to assist in the constraint solving process, and results showing why they should still be taken into account in parallel constraint solving.

Keywords: Constraint solving, Parallelism, GPU, Intel MIC, Heterogeneous systems

1 Introduction

Constraint Satisfaction Problems (CSPs) allow modelling problems like the Costas Array problem [6], and some real life problems like planning and scheduling [2], resources allocation [7] and route definition [3].

CPU's parallelism is already being used with success to speed up the solving processes of harder CSPs [5,12,14,16]. However, very few constraints solvers contemplate the use of GPUs. In fact, Jenkins *et al.* recently concluded that the execution model and the architecture of GPUs are not well suited to computations displaying irregular data access and code execution patterns such as backtracking search [9].

We are currently developing a constraint solver named Parallel Heterogeneous Architecture Toolkit (PHACT) that is already capable of achieving state-of-the-art performances on multi-core CPUs, and can also speed up the solving process

by adding GPUs and processors like Intel Many Integrated Cores (MICs) to solve the problems.

The next section introduces the main CSP concepts and Section 3 presents some related work. Section 4 describes the architecture of PHACT, and in Section 5 the results achieved with PHACT, when solving some CSPs on multiple combinations of devices and when compared with some state-of-the-art solvers, are presented and discussed. Section 6 presents the conclusions and directions for future work.

2 CSPs concepts

A CSP can be briefly described as a set of variables with finite domains, and a set of constraints between the values of those variables. The solution of a CSP is the assignment of one value from the respective domain to each one of the variables, ensuring that all constraints are met.

For example, the Costas Array problem consists in placing n dots on a $n \times n$ matrix such that each row and column contain only one dot and all vectors between dots are distinct. It can be modelled as a CSP with $n + n(n-1)/2$ variables, n of which correspond to the dots and each one is mapped to a different matrix column. The domain of these n variables is composed by the integers that correspond to the matrix rows where each dot may be placed. The remaining $n(n-1)/2$ variables constitute a difference triangle, whose rows cannot contain repeated values [6].

The methods for solving CSPs can be categorized as incomplete or complete. Incomplete solvers do not guarantee that an existing solution will be found, being mostly used for optimization problems and for large problems that would take too much time to fully explore. Incomplete search is beyond the scope of this paper and will not be discussed here. On the contrary, complete methods guarantee that if a solution exists, it will be found.

3 Related work

Searching for CSP solutions in a backtracking approach can be represented in the form of a search tree. To take advantage of parallelism this search tree may be split into multiple subtrees and each one of them explored in a different thread that may be running on a different core, device or machine. This is the approach generally found in parallel constraint solvers, which run on single or distributed multi-core CPUs [5,12,14,16].

Pedro developed a CSP solver named Parallel Complete Constraint Solver (PaCCS) capable of running from a single core CPU to multiple multi-core CPUs in a distributed system [12]. Using work stealing for distributing the work among the threads and the Message Passing Interface (MPI) to allow communication between them, this solver achieved almost linear speedups for most of the problems tested.

Régin *et al.* implemented an interface responsible for decomposing an initial problem into multiple sub-problems, filtering out those found to be inconsistent [15]. After generating the sub-problems it creates multiple threads, each one corresponding to an execution of a solver (e.g., Gecode [17]), to which a sub-problem is sent at a time for exploration.

For some optimization and search problems, where the full search space is explored, these authors achieved average gains of 13.8 and 7.7 against a sequential version, when using Gecode through their interface or just Gecode, respectively [15]. On their trials, the best results were achieved when decomposing the initial problem into 30 sub-problems per thread and running 40 threads on a machine with 40 CPU cores.

While solving CSPs through parallelization has been a subject of research for decades, the usage of GPUs for that purpose is a recent area, and as such there aren't many published reports of related work. To our knowledge, there are only two published papers related with constraint solving on GPUs [1,4]. From these two, only Campeotto *et al.* presented a complete solver [4].

Campeotto *et al.* developed a CSP solver with Nvidia's Compute Unified Device Architecture (CUDA), capable of using simultaneously a CPU and an Nvidia GPU to solve CSPs [4]. On the GPU, this solver implements an approach different from the one mentioned before, namely, instead of splitting the search tree over multiple threads, it splits each constraint propagation over multiple threads, and exploits parallelism on three levels:

- Constraints relating many variables are propagated on the GPU, while the remaining constraints are filtered sequentially by the CPU.
- On the GPU, the propagation and consistency check for each constraint is assigned to one or more blocks of threads according to the number of variables involved;
- The domain of each variable is filtered by a different thread.

Campeotto *et al.* reduced the data transfer to a minimum by transferring to the GPU only the domains of the variables that weren't labelled yet and the events generated during the last propagation. Events identify the changes that happened to a domain, like becoming a singleton or having a new maximum value, which allows deciding on the appropriate propagator to apply.

All the data transfers between host and device are made asynchronously, and only after the CPU has finished his sequential propagation do the GPU and CPU synchronize.

Campeotto *et al.* obtained speedups of up to 6.61, with problems like the Langford problem and some real problems such as the modified Renault problem [10], when comparing a sequential execution on a CPU with the hybrid CPU/GPU version.

4 Solver architecture

PHACT is a complete solver, capable of finding a solution for a CSP if one exists. It is meant to be able to use all the (parallel) processing power of the devices

available on a system, such as CPUs, GPUs and MICs, to speed up solving constraint problems.

The solver is composed of a master process, which coordinates the search, and of (multithreaded) solving agents, where the search for solutions takes place. A search-space splitting strategy is employed in the parallelization of the search process. The master process is responsible for the splitting and for distributing work to the agents, and each agent's thread implements a search engine, performing labelling, constraint propagation and backtracking on one sub-search space at a time.

PHACT may be used to count all the solutions of a given CSP, to find just one solution or a best one (for optimization problems).

Framework

PHACT is implemented in C and OpenCL [11], which allows its execution on multiple types of devices from different vendors and the capability of being executed on Linux or on Microsoft Windows.

We present some OpenCL concepts, in order to better understand PHACT's architecture:

- **Compute unit** One or more processing elements and their local memory. In Nvidia GPUs each Streaming Multiprocessor (SM) is a compute unit. AMD GPUs have their own components called Compute Units that match this definition. For CPUs and MICs, the number of available compute units is normally equal to or higher than the number of threads that the device can execute simultaneously [11];
- **Kernel** The code that will be executed on the devices;
- **Work-item** An instance of the kernel (thread);
- **Work-group** Composed of one or more work-items that will be executed on the same compute unit, in parallel. All work-groups for one kernel on one device have the same number of work-items;
- **Host** CPU where the application responsible for managing the execution of the kernels is run;
- **Device** A device where the kernels are executed (CPU, GPU, MIC).

In the implementation described here, the master process runs on the OpenCL host and the agents run on the devices. An agent consists of one or more work-groups, each having one or more work-items, and all work-items execute the same kernel code, which implements the search engine.

Search space splitting and work distribution

For distributing the work between the devices, PHACT splits the search space into multiple sub-search spaces. Search-space splitting is effected by partitioning the domains of one or more of the problem's variables, so that the resulting sub-search spaces partition the full search space. The number and the size of the

sub-search spaces thus created depend on the number of work-items which will be used.

Example 1 shows the result of splitting the search space of a CSP with three variables, $V1$, $V2$ and $V3$, all with domain $\{1, 2\}$, into 4 sub-search spaces, $SS1$, $SS2$, $SS3$ and $SS4$.

Example 1. Creation of 4 sub-search spaces.

$$SS1 = \{V1 = \{1\}, V2 = \{1\}, V3 = \{1, 2\}\}$$
$$SS2 = \{V1 = \{1\}, V2 = \{2\}, V3 = \{1, 2\}\}$$
$$SS3 = \{V1 = \{2\}, V2 = \{1\}, V3 = \{1, 2\}\}$$
$$SS4 = \{V1 = \{2\}, V2 = \{2\}, V3 = \{1, 2\}\}$$

Since each device will have multiple search engines running in parallel, the computed partition is organized into blocks of contiguous search spaces (according to some enumeration of the search spaces).

The process running on the host launches the execution of the solving agents on the devices, hands each device one block of sub-search spaces to explore, and coordinates the progress of the solving process as each device finishes exploring its assigned block. The coordination of the devices consists in assessing the state of the search, distributing more work to the devices, signalling to all the devices that they should stop (when a solution has been found and only one is wanted), or updating the current bound (in optimization problems).

Load balancing

An essential aspect to consider when parallelizing some task is the balancing of the work between the parallel components. Creating sub-search spaces with balanced domains, when possible, is no guarantee that the amount of work involved in exploring each of them is the same, or even similar. To compound the issue, we are dealing with devices with differing characteristics and varying speeds, making it even harder to statically determine an optimal, or even good, work distribution.

Achieving effective load balancing between devices with such different architectures as CPUs and GPUs is a complex task [9]. When trying to implement dynamic load balancing, two important OpenCL limitations arise, namely when a device is executing a kernel it is not possible for it to communicate with other devices [8], and the execution of a kernel can not be paused or stopped. Hence, some techniques like work stealing [5,13], which requires communication between threads, will not work with kernels that run independently on different devices and load balancing must be done on the host side.

To better manage the distribution of work, the host could reduce the amount of work it sends to the devices each time, by reducing the number of search spaces in each block. This would make the devices synchronize more frequently on the host and allow for a finer control over the behaviour of the solver. When working with GPUs, though, the number and the size of data transfers between devices and host should be as small as possible, because these are very time consuming

operations. So, a balance must be struck between the workload of the devices and the amount of communication needed.

PHACT implements a dynamic load balancing technique which adjusts the size of the blocks of sub-search spaces to the performance of each device solving the current problem, when compared to the performance of the other devices.

Initially each device d explores a small block of sub-search spaces to get the *average time*, $avg(d)$, it needs to explore one sub-search space. When two or more devices finish exploring that first block, their *rank*, $rank(d)$, which consists of a value between 0 and 1, corresponding to the relative speed of device d against all the devices that were used for solving a block of sub-search spaces, is calculated according to the Equation 1, where m is the total number of devices.

$$rank(d) = \frac{\frac{1}{avg(d)}}{\sum_{i=1}^{m} \frac{1}{avg(i)}}, \quad avg(i) > 0 \tag{1}$$

The rank of a device estimates the fraction of the search space it could explore in the time the remaining devices would need to explore the rest of the search space. Since the first block of sub-search spaces explored by each device is small, to prevent slow devices from dominating the solving process, it often only allows for a rough approximation of the speed of a device. So, in the beginning, only a part of the remaining search spaces is considered when computing the size of the next block to send to a device.

As search progresses, every time a device finishes exploring another block, its rank is updated. As its value stabilizes, the size of the new block of search spaces for the device will be the corresponding percentage from all unexplored sub-search spaces. In the end, when the device waiting for work is estimated to need less than one second to solve all the remaining sub-search spaces, it will be assigned all of them.

If a device receives less sub-search spaces than the amount needed to fully exploit its level of parallelism, that device can apply a multiplier factor m to the size of a block and turn a block of sub-search spaces into a block with m times the original number of search spaces. This technique allows reducing the variation of avg when working with small blocks.

Communication

To reduce the amount of data that is transferred to each device, all of them will receive the full CSP, that is, all the constraints, variables and their domains, at the beginning of the solving process. Afterwards, when a device must be instructed to solve a new block of sub-search spaces, instead of sending all the sub-search spaces to the device, only the information needed to create those sub-search spaces is sent.

If a device is to solve sub-search spaces $SS2$ and $SS3$ from Example 1, it will receive the information that the tree must be expanded down to depth 2, that the values of the first variable are repeated 2 times and the values of the

second variable are repeated 1 time only (not repeated). With this information the device will know that the values of the first variable are repeated 2 times, so the third sub-search space ($SS3$) will get the second value of that variable, and so on to the expansion depth. The values of the variables that were not expanded are simply copied from the original CSP that was passed to the devices at the beginning of the solving process.

PHACT represents the variable domains as bitmaps. When solving the n-Queens problem with 17 queens a 32-bit bitmap is used for the domain of each of the 17 variables. If 200,000 sub-search spaces were to be created, that would lead to a search tree expansion depth of 5. As such, at least these five levels would be created for each sub-search space resulting in 1,000,000 bitmaps ($200,000 \times 5$) of 32-bits each (32,000,000 bits) that would have to be stored in the host memory and transferred to the device memory.

With this technique, the 32 Mb are replaced just by the information needed for the work-items to generate the sub-search spaces, and that is also what is sent to each device instead of blocks of fully created sub-search spaces.

5 Results and discussion

PHACT was evaluated on finding all the solutions for the Costas Array problem with 14 dots and the n-Queens problem with 17 queens. Those tests were executed on one, two and three devices and on four different machines running Linux to evaluate the speedups when adding more devices to help the CPU.

PHACT performance was also compared with those of PaCCS and Gecode on these four machines.

The four machines have the following characteristics:

- Machine with 32 GB of RAM (referred to as M1 in the remainder of this paper) and:
  - Intel Core i7-4870HQ (referred to as I7, with 8 compute units);
  - Nvidia GeForce GTX 980M (Geforce, 12 compute units).
- Machine with 64 GB of RAM (M2) and:
  - Intel Xeon E5-2690 v2 (Xeon 1, 40 compute units);
  - Nvidia Tesla K20c (Tesla, 13 compute units).
- Machine with 128 GB of RAM (M3) and:
  - AMD Opteron 6376 (Opteron, 64 compute units);
  - Two AMD Tahitis (Tahiti 1 and Tahiti 2, 32 compute units each). These two devices are combined in an AMD Radeon HD 7990, but are managed separately by OpenCL.
- Machine with 64 GB of RAM (M4) and:
  - Intel Xeon CPU E5-2640 v2 (Xeon 2, 32 compute units);
  - Two Intel Many Integrated Core 7120P (MIC 1 and MIC 2, 240 compute units each).

Machine	Devices	Elapsed time (s)	Speedup vs. CPU
M1	Geforce	164,0	0.73
	I7	118,9	1.00
	Geforce and I7	77,4	1.54
M2	Tesla	957,8	0.03
	Xeon 1	28,6	1.00
	Tesla and Xeon 1	28,7	1.00
M3	Tahiti 1	359,9	0.07
	Tahiti 2	360,0	0.07
	Opteron	26,2	1.00
	Tahiti 1 and Opteron	25,3	1.04
	Tahiti 2 and Opteron	25,3	1.03
	Tahiti 1, Tahiti 2 and Opteron	24,5	1.07
M4	MIC 1	93,7	0.62
	MIC 2	93,7	0.62
	Xeon 2	57,7	1.00
	MIC 1 and Xeon 2	39,0	1.48
	MIC 2 and Xeon 2	39,0	1.48
	MIC 1, MIC 2 and Xeon 2	30,2	1.91

Table 1. Elapsed times and speedups for finding all the solutions for the Costas Array with 14 dots.

Table 1 presents the elapsed times on all these machines and devices when finding all the solutions for the Costas Array with 14 dots and the speedups achieved when using the CPU together with the other devices over using only the CPU.

For this problem, all the GPUs were outperformed by the CPUs from the same machine. The best GPU performance was achieved with the Geforce which was about 38% slower than I7. However, adding Geforce to I7 allowed to reduce the running time in about 34%, resulting in a speedup of 1.54.

Tesla was the GPU with the worst performance in all the tests. For this problem it was about 33 times slower than the CPU of this machine (M2). Because of these differences in performance, when adding Tesla to help Xeon 1, the elapsed timed did not decrease, in fact it increased a little. This was due to the fact that the work spared the Xeon 1 by Tesla solving some sub-search spaces did not make up for the extra work that Xeon 1 (host) had to control Tesla (device).

On M3, Tahiti 1 and Tahiti 2 allowed to speed up the solving process in 1.07 when compared with using only the Opteron CPU. This may seem a small speedup, but we must note that Opteron was the fastest CPU in all the tests.

On M4, adding just one MIC to help Xeon 2 allowed a speedup of 1.48 and when adding both MIC 1 and MIC 2 allowed a speedup of 1.91, reducing the elapsed time to almost half than when using only the CPU.

Table 2 presents the same comparisons as Table 1, but for finding all the solutions for the n-queens with 17 queens.

For this problem, Geforce alone was actually more than 2 times faster than I7 alone. All the other GPUs also achieved better speedups with the n-Queens than with the Costas Array, when compared with the CPUs elapsed times. This fact may be explained by the direct relation between the size of local memory

Machine	Devices	Elapsed time (s)	Speedup vs. CPU
M1	Geforce	120,4	2.23
	I7	268,4	1.00
	Geforce and I7	93,2	2.88
M2	Tesla	541,1	0.11
	Xeon 1	61,0	1.00
	Tesla and Xeon 1	59,9	1.02
M3	Tahiti 1	146,2	0.40
	Tahiti 2	146,2	0.40
	Opteron	59,1	1.00
	Tahiti 1 and Opteron	47,1	1.25
	Tahiti 2 and Opteron	47,2	1.25
	Tahiti 1, Tahiti 2 and Opteron	41,2	1.43
M4	MIC 1	187,2	0.57
	MIC 2	187,3	0.57
	Xeon 2	107,3	1.00
	MIC 1 and Xeon 2	71,8	1.49
	MIC 2 and Xeon 2	71,8	1.49
	MIC 1, MIC 2 and Xeon 2	56,1	1.91

Table 2. Elapsed times and speedups for finding all the solutions for the n-Queens with 17 queens.

that each work-item requires and the number of variables of the CSP. The more variables a CSP has, the more local memory each work-item will require, and less work-items per work-group can be effectively used.

The Costas Array with 14 dots that was modelled in PHACT uses 119 variables and n-Queens with 17 queens uses only 17 variables, which results in GPUs being capable of efficiently using about 7 times more work-items per work-group for n-Queens than for the Costas Array, allowing for much better performances.

On M4, the speedups were very similar to the ones achieved for the Costas Array, because in PHACT both CPUs and MICs use only one work-item per work-group, as each CPU and MIC compute unit can only execute one thread at a time.

The comparison between the elapsed times for PHACT, Gecode and PaCCS on finding all the solutions for the Costas Array with 14 dots and the n-Queens with 17 queens is presented in Table 3.

The three solvers were executed on the CPUs of the 4 machines using only one compute unit with one thread/work-item (column "1 CPU CU"), and the number of threads/work-items that allowed to achieve the best results (column "All CPU CUs"). PaCCS used the same number as compute units and PHACT used 1,024 work-items on each CPU. Gecode was executed on I7 with 8 threads (same as the number of compute units), but with half the number of compute units on the other 3 CPUs, because it was noted that a higher number of threads on this CPUs lead to an increased elapsed time. Of the three solvers only PHACT is capable of using GPUs and MICS, so PHACT was the only solver that was also executed using all the compute units of all the available devices on each machine (column "All CUs").

CSP	Machine	Elapsed times (s)						
		Gecode		PaCCS		PHACT		
		1 CPU CU	All CPU CUs*	1 CPU CU	All CPU CUs	1 CPU CU	All CPU CUs	All CUs
Costas Array 14	M1	1292,7	305,2	741,0	177,7	507,9	118,9	77,4
	M2	1442,6	86,4	880,6	39,4	578,9	28,6	28,7
	M3	2586,9	283,8	1700,6	38,4	999,7	26,2	24,5
	M4	2073,3	444,0	1274,1	70,1	903,3	57,7	30,2
N-Queens 17	M1	1647,4	479,8	2219,4	523,0	1290,9	268,4	93,2
	M2	1789,7	778,4	2454,4	149,0	1427,1	61,0	59,9
	M3	2968,4	1325,8	4606,0	101,3	2336,7	59,1	41,2
	M4	2556,4	1003,3	3546,9	233,2	2094,4	107,3	56,1

Table 3. Comparison of the elapsed times for Gecode, PaCCS and PHACT when finding all the solutions for the Costas Array 14 and the n-Queens 17.

All the solvers used the first-fail[1] heuristic to select the variable to label and the smallest value of the variable domain to assign.

PHACT achieved the best results of the three solvers for all the tests.

Table 4 presents the speedups relative to the elapsed times presented in Table 3.

CSP	Machine	Speedup PHACT					
		vs. Gecode			vs. PaCCS		
		1 CPU CU	All CPU CUs	All CUs*	1 CPU CU	All CPU CUs	All CUs*
Costas Array 14	M1	2.55	2.57	3.95	1.46	1.49	2.30
	M2	2.49	3.02	3.01	1.52	1.38	1.37
	M3	2.59	10.83	11.58	1.70	1.47	1.57
	M4	2.30	7.69	14.70	1.41	1.21	2.32
N-Queens 17	M1	1.28	1.79	5.15	1.72	1.95	5.61
	M2	1.25	12.76	12.99	1.72	2.44	2.49
	M3	1.27	22.43	32.17	1.97	1.71	2.46
	M4	1.22	9.35	17.89	1.69	2.17	4.16

Table 4. Speedups achieved by PHACT when compared with Gecode and PaCCS for the elapsed times presented in Table 3.

PHACT achieved speedups that ranged from 1.22 to 2.59 when using a single work-item on the CPUs, against Gecode and PaCCS when using only one thread.

When compared with Gecode, the speedups ranged from 1.22 when using a single core of Xeon 2 to 22.43 when using all the cores on Opteron. When using all the devices, PHACT achieved speedups of up to 32.17 when compared with the best results on the same machine for Gecode. With the exception of the speedup for the Costas Array on Xeon 1, we can see that the gap between Gecode and PHACT increases as the number of CPU compute units used also increases.

When compared with PaCCS, the speedups ranged from 1.21 on Xeon 2 to 2.44 on Xeon 1 when using all the CPU cores, which shows that PHACT is

[1] Selecting the variable with less values left in its domain.

capable of better harnessing the parallel processing power of the used CPUs for the solved CSPs. If using all the devices, PHACT achieved speedups of up to 5.61 when compared with PaCCS.

However, it must be noted that PHACT is yet being developed and only a few propagators are already implemented. More propagators will be implemented and that may influence the results.

Figure 1 represents the elapsed times achieved with Gecode, PaCCS and PHACT when using all the available compute units of the four machines, that each solver is capable of using, for finding all the solutions for the Costas Array with 14 dots (left chart) and the n-Queens with 17 queens (right chart).

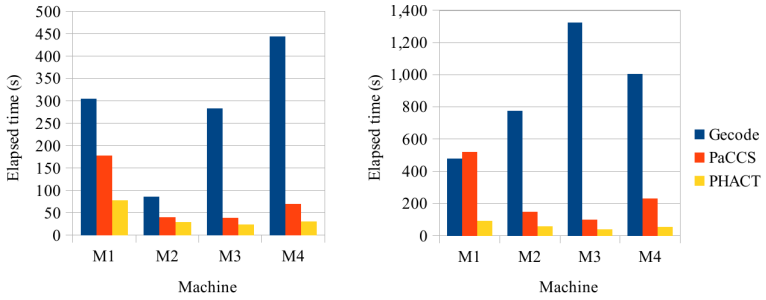


Fig. 1. Elapsed times for Gecode, PaCCS and PHACT when finding all the solutions for the Costas Array 14 and the n-Queens 17 using all compute units each solver can use.

Although PHACT was faster than Gecode and PaCCS on all the results that were presented in this section, PHACT is also capable of using GPUs and MICs to increase its performance even further.

6 Conclusion and future work

To our knowledge, PHACT is the only constraint solver capable of using simultaneously CPUs, GPUs, MICS and any other device compatible with OpenCL to solve CSPs in a faster manner.

Although GPUs are not particularly efficient for this type of problems, they can speed up the solving process and in some cases be even faster than the CPU of the same machine. PHACT achieved speedups of up to 2.88 when using a GPU to help the CPU on the same machine, and 1.91 when aided by two MICs. These results achieved with GPUs are much better than the ones presented by Jenkins *et al.* in [9] which concluded that a GPU was only capable of being 1.4 to 2.25 times as fast as a single CPU core when exploring the backtracking paradigm.

PHACT performance was also compared with the ones of Gecode and PaCCS, achieving speedups that ranged from 1.22 to 22.43 when compared with Gecode

and 1.21 to 2.44 when compared with PaCCS, which shows that PHACT is better on harnessing the multi-core CPUs processing power to solve the presented CSPs.

When using CPUs, GPUs and MICs to speed up the solving process, PHACT achieved speedups of up to 32.17 when compared to Gecode and of up to 5.61 when compared to PaCCS. However, Gecode and PaCCS can only use the CPUs.

Currently PHACT is already being extended with more propagators that will allow it to solve many other CSPs and in the near future PHACT should also be extended to work on distributed environments.

Acknowledgments

This work was partially funded by Fundação para a Ciência e Tecnologia (FCT) under grant UID/CEC/4668/2016 (LISP). Some of the experimentation was carried out on the `khromeleque` cluster of the University of Évora, which was partly funded by grants ALENT-07-0262-FEDER-001872 and ALENT-07-0262-FEDER-001876.

References

1. Arbelaez, A., Codognet, P.: A GPU implementation of parallel constraint-based local search. In: 22nd Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP). pp. 648–655. PDP '14, IEEE, Italy (2014)
2. Barták, R., Salido, M.A.: Constraint satisfaction for planning and scheduling problems. Constraints 16(3), 223–227 (July 2011)
3. Brailsford, S.C., Potts, C.N., Smith, B.M.: Constraint satisfaction problems: Algorithms and applications. European Journal of Operational Research 119(3), 557–581 (1999)
4. Campeotto, F., Palù, A.D., Dovier, A., Fioretto, F., Pontelli, E.: Exploring the use of GPUs in constraint solving. In: Flatt, M., Guo, H.F. (eds.) Sixteenth International Symposium on Practical Aspects of Declarative Languages (PADL 2014). LNCS, vol. 8324, pp. 152–167. San Diego, CA, USA (January 2014)
5. Chu, G., Schulte, C., Stuckey, P.J.: Confidence-based work stealing in parallel constraint programming. In: Gent, I.P. (ed.) The 15th International Conference on Principles and Practice of Constraint Programming. LNCS, vol. 5732, pp. 226–241. Springer, Lisbon, Portugal (September 2009)
6. Diaz, D., Richoux, F., Codognet, P., Caniou, Y., Abreu, S.: Constraint-based local search for the costas array problem. In: Hamadi, Y., Schoenauer, M. (eds.) Learning and Intelligent Optimization: 6th International Conference, (LION 6). LNCS, vol. 7219, pp. 378–383. Springer (2012)
7. Filho, C., Rocha, D., Costa, M., Albuquerque, P.: Using constraint satisfaction problem approach to solve human resource allocation problems in cooperative health services. Expert Syst. Appl. 39(1), 385–394 (2012)
8. Gaster, B., Howes, L., Kaeli, D.R., Mistry, P., Schaa, D.: Heterogeneous Computing with OpenCL. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1st edn. (2011)

9. Jenkins, J., Arkatkar, I., Owens, J., Choudhary, A., Samatova, N.: Lessons learned from exploring the backtracking paradigm on the GPU. In: Jeannot, E., Namyst, R., Roman, J. (eds.) Euro-Par 2011 Parallel Processing, LNCS, vol. 6853, pp. 425–437. Springer Berlin Heidelberg (2011)
10. Mairy, J.B., Deville, Y., Lecoutre, C.: Integration of AI and OR Techniques in Constraint Programming: 11th International Conference, CPAIOR 2014, Cork, Ireland, pp. 235–250. Springer International Publishing (2014)
11. Munshi, A., Gaster, B., Mattson, T.G., Fung, J., Ginsburg, D.: OpenCL Programming Guide. Addison-Wesley Professional, 1st edn. (2011)
12. Pedro, V.: Constraint Programming on Hierarchical Multiprocessor Systems. Ph.D. thesis, Universidade de Évora (2012)
13. Pedro, V., Abreu, S.: Distributed work stealing for constraint solving. In: Vidal, G., Zhou, N.F. (eds.) Joint Workshop on Implementation of Constraint Logic Programming Systems and Logic-based Methods in Programming Environments (CICLOPS-WLPE 2010). Edinburgh, Scotland, U.K. (July 2010)
14. Rolf, C.C., Kuchcinski, K.: Parallel solving in constraint programming. In: Third Swedish Workshop on Multi-Core Computing (MCC 2010) (November 2010)
15. Régin, J.C., Rezgui, M., Malapert, A.: Embarrassingly parallel search. In: Schulte, C. (ed.) Principles and Practice of Constraint Programming. NCS, vol. 8124, pp. 596–610. Springer Berlin Heidelberg (2013)
16. Schulte, C.: Parallel search made simple. In: Beldiceanu, N., Harvey, W., Henz, M., Laburthe, F., Monfroy, E., Müller, T., Perron, L., Schulte, C. (eds.) Proceedings of TRICS: Techniques foR Implementing Constraint programming Systems, a postconference workshop of CP 2000. Singapore (September 2000)
17. Schulte, C., Duchier, D., Konvicka, F., Szokoli, G., Tack, G.: Generic constraint development environment. `http://www.gecode.org/`

Information Extraction from Microblogs

Ganchimeg Lkhagvasuren
Evora University
ganchimeg@seas.num.edu.mn

ABSTRACT

This paper presents several results of our experiment on a dataset released in Microblog task of Forum for Information Retrieval Evaluation - 2016 (FIRE2016). We combine Word embeddings (WE), Wordnet (Wnet) and BM25 for both processes of the query expansion and the retrieving. Our ensemble approach consists those methods achieves a 5.71% increase in Precision at the top-ranked 20 documents and a 11.08% increase in Recall at the top-ranked 1000 documents.

CCS Concepts

• **Information systems→**Information Extraction.

Keywords

Information extraction; Word Embeddings, Query expansion, Wordnet, BM25, Part-of-Speech

1. INTRODUCTION

It is undeniable that microblogging sites have become key resources of significant information during disaster event [1]. One of these microblogging site, Twitter, is a social networking website which enables users to generate 140-character messages named "tweets". Everyday, A giant number of tweets is posted including informative and non-informative massages, which makes opportunities for information extraction [3].

However, dealing with tweets and identifying specific keywords are challenging work due to the nature of Twitter. The small, noisy and fragmented *tweets means they have very simple discourse and pragmatic structure, issues which still challenge state-of-the-art NLP systems* [2].

All methods attempted to the FIRE2016 used only one methods for all seven topics expect introduced Named Entity Recogniser (NER) to Topic 5 and 6. In this paper, we proposed to compare the methods performance for each of the topics and build an ensemble system based on the best methods.

In terms of our experiment, we proposed to compare some techniques such as query expansions, scoring models and so on, to see which method performs better for each of the given seven topics. Based on our experiments, we build a system that contains ensemble methods for every topics in the task.

This paper organized as follows. First, related work in IE and approaches used in the task are described. Then, the result analysis and conclusion are presented.

2. RELATED WORK

In the FIRE2016, there were few automatic systems that attempts to extract information. Reference [] used Wordnet to expand keywords and then employed BM25 retrieval method. Similarly, reference [16] and [17] extended keywords by Wordnet and then extracted relevant tweets by calculating similarity of sentences and queries based on WE models such as Word2Vec model.

In this paper, WE, Wnet and BM25 methods are used in both of query expansion and retrieval phases. First of all, we extract seed keywords by Part-of-Speech tagger for our specific IE task. When it comes to expanding seed keywords we employ word2vec model to get clusters of seed keywords. And when it comes to retrieving tweets, all tweets and queries are calculated by cosine similarity measurement based on their WE and then ranked. From the results of performances, we create ensemble approach based on which approach is performed better on each topic.

Unlike previous works that use only one method for all seven topics of the task, we attempt to make ensemble methods.

3. APPROACH

We begin this section by summarizing details of the selecting seed words from the topics, then describe how we expand those words and some methods that we employed in this task.

3.1 Extract seed keywords

Since the given topics plays a crucial role in this task, first we extracted seed keywords from the titles, the descriptions and narratives of the topics with stopwords. Nouns contain bulk of information rather the other part speeches [8]. Therefore when we were extracting the seed words from the topics, we had a question that how efficient is if we use only nouns or nouns with verbs as long to filter unlikely keywords. Then we did a little experiment, first we tagged each topic by Stanford Part-of-speech tagger [], then we evaluated the results on the topics by using only nouns, nouns with adjectives and so on, as the keywords. Experimental setup is same as the setup described in section 4. The results (Table 1) demonstrates that using all filtered words after applying stopwords to the topics is not efficient rather using only the nouns or the nouns with the adjectives as the seed words in the task. According to the result, we selected nouns and adjectives for our task.

Table 1. Results the parts of speech as seed words in search engine

Run_ID	Precision @ 20	Recall @ 1000
All words + stopwords	0,3571	0,37
Nouns	0,3714	**0,4085**
Nouns + Verbs	0,3571	0,38
Nouns + Adjectives	**0,3785**	0,4042
Nouns + Verbs + Adjectives	0,3642	0,3771

3.2 Query Expansion

Query Expansion (QE) a well-known technique that attempts to increase the likelihood of a match between the query and relevant documents reformulating seed query terms. Many researchers have explored a variety of automatic query expansion techniques for collecting these terms and its comprehensive survey can be found on [10].

The specific requirement of the queries for instance, "G*eneralized statements without reference to any resources or messages asking for donation of money would be relevant*", requires rational and wide keywords. This feature makes inferior results (see Table 3) on Relevance Feedback based QE techniques which are widely used to retrieve more relevant documents and improve overall performance. Retrieving documents that do not contain any of the query terms but relevant to the query and not general, is necessary in our task. Therefore we considered the following two External based resources to expand the seed keywords.

Wordnet (Wnet)

A QE technique is that to expand seed keywords using their synonyms (and possibly holonyms, meronyms, etc.). The information of synonyms can be extracted from Wordnet[1] which is a lexical database for the English language. Using Wordnet for QE and to disambiguate the sense of query words already showed considerable results in some researches [][]. However an issue, if a keyword occurs in multiple synsets, which synset(s) should be selected, is needed to be addressed when Wordnet is used. In our task, we enriched the seed keywords by all synonyms from all the Wordnet synsets in which keywords occur.

Word Embeddings (WE)

Another alternative QE technique is Word Embeddings (WE) that attempts to expand the seed keywords by selecting their cluster words. When expanding the seed keywords based on WE, there are two ways to train data: locally and globally. The study [11] shows that selecting terms from locally trained WE is better than globally trained WE for QE. In our task, we selected nearest k terms to the initial every query terms based on measuring their representations obtained from a Word2vec model trained on a huge Twitter data [12] by cosine similarity (Equation 1). First we removed all stopwords from the dataset provided and created a bag of words. Then every pair between query terms and the bag of words is computed by cosine similarity.

3.3 Retrieval model based on Word Embeddings

An another technique for information extraction is using Word embedding to measure how relevant the query terms to the tweets based on word representations. This method were used by some other participants [16][17] of FIRE2016 and showed relatively better results. The idea of this method is first to build a vector for every single tweet and query terms, and then to calculate between document vectors and query term vectors. To convert documents and query terms to vectors of 200 dimensions, we used Word2Vec model [5] which was trained particularly a huge tweet data in order to recognize NER for Twitter.

In our experiment, the tweet vectors were built by taking the normalized summation of the vectors of the words in the tweets after stopwords and urls were removed in the preprocessing. In cases where the word is not a contained in the model vocabulary, it is assigned to the null vector. Similarly, each topics word bag vectors were computed. The similarity between vectors of tweets and query terms are calculated by cosine.

3.4 Named Entity Recognition

FMT5 requires to identify tweets inform locations where resources were available and required whereas FMT6 requires to detect tweets inform organizations were active during the disaster. Although a named entity recogniser fails to identify locations and organizations due to complication of Twitter. Most of identified locations and organizations names do not belong to the places and organizations of Nepal. Therefore we used lists of location and organization of Nepal from external resources. A list of cities of Nepal can be found from Wikipedia [13] while a bunch of global non-government organization and government organization of Nepal are listed in [14] and [15] separately.

4. EXPERIMENT

Task description: The FIRE 2016 Microblog track [7] called for the extraction of relevant tweets for 7 topics (Appendix A) in TREC format from within a large number of tweets posted during the Nepal earthquake in April, 2015. Each topic, comprises from a title, a brief description and a narrative, requires a broad information such as availability and requirement of resources, locations and so on .

This is indisputably an ad-hoc search task and participants were challenged to deal with small, noisy tweets and identifying specific keywords for each topic with high precision as well as high recall.

Dataset: Approximately 50,000 tweets that posted during Nepal earthquake disaster in English are given in JSON[2] format. The tweets were collected using the Twitter Search API[3] and the keyword "nepal" [7]. Yet the value of information that is conveyed by tweets varies greatly, some of tweets contain important messages whereas some of tweets are more personal.

Evaluation: The gold standard provided is developed by human annotators who is proficient in English and a normal user of Twitter. It contains the following number of tweets judged relevant to the seven topics - FMT1: 589, FMT2: 301, FMT3: 334, FMT4: 112, FMT5: 189, FMT6: 378, FMT7: 254. Also the following measures are used to evaluate results: (i) Precision at 20 (Prec@20), Recall at 1000 (Recall@1000) and Mean Average Precision at 1000 (MAP@1000). In these measures, at first, all attempts are ranked by their precisions, if they have same precisions then MAP and recall measures are considered respectively.

[1] https://wordnet.princeton.edu/

[2] www.json.org

[3] www.dev.twitter.com/rest/public/search

Table 2. The experimental results

	Average		Topic1		Topic2		Topic3		Topic4		Topic5 + LOC		Topic6 + ORG		Topic7	
	P@20	R@1000	P@20	R@1000	P@20	R@1000	P@20	R@1000	P@20	R@1000	P@20	R@1000	P@20	R@1000	P@20	R@1000
QT + BM25	0.41	0.41	0.2	0.37	0.5	0.65	0.4	0.4	0.5	0.55	0.5	0.32	0.15	0.12	0.65	0.49
QT + WE(Cosine) + GT	0.19	0.22	0.3	0.22	0.35	0.29	0.2	0.3	0.35	0.35	0	0.14	0.1	0.11	0.05	0.1
QT + WE(Cosine) + LT	0.37	0.35	0.4	0.42	0.4	0.5	0.45	0.4	0.35	0.45	0.3	0.24	0.2	0.11	0.5	0.33
QT+QE(Wnet) + BM25	0.31	0.32	0.1	0.19	0.5	0.46	0.55	0.45	0.3	0.42	0.3	0.34	0.1	0.08	0.3	0.33
QT+QE(Wnet) + WE(Cosine) + GT	0.33	0.35	0.2	0.26	0.5	0.5	0.45	0.42	0.3	0.4	0.35	0.31	0.15	0.2	0.35	0.35
QT + QE(Wnet) + WE(Cosine) + LT	0.38	0.37	0.25	0.31	0.45	0.51	0.45	0.44	0.5	0.41	0.4	0.29	0.15	0.2	0.45	0.4
QT+QE(WE)+ BM25	0.41	0.39	0.35	0.4	0.5	0.55	0.4	0.38	0.5	0.52	0.5	0.32	0.15	0.11	0.45	0.48
QT+QE(WE) + WE(Cosine) + GT	0.19	0.21	0.45	0.21	0.2	0.3	0.35	0.28	0.2	0.33	0.05	0.16	0.1	0.07	0	0.1
QT + QE(WE) + WE(Cosine) + LT	0.47	0.40	0.45	0.41	0.5	0.4	0.55	0.5	0.5	0.55	0.5	0.32	0.3	0.22	0.5	0.42
Overall best	0.49	0.45	0.45	0.42	0.5	0.65	0.55	0.5	0.5	0.55	0.5	0.34	0.3	0.22	0.65	0.49

We used WE in two processes namely: Word Expansion and Retrieval. In the former process, we calculate a cluster of each keywords and then pick nearest 10 words with that similarity is over than 0.5.

Moreover we used the TERRIER[4] open source retrieval system for our experiment. At the time of indexing, stopwords are removed and Porter stemmer is employed and no phrases are used in preprocessing.

5. RESULT AND DISCUSSION

The experimental results are shown in Table 2. In the result table QT, GT and LT refer to query terms, global train and local train respectively. The results shows that the best results of Topic 1,3,4 and 6 were achieved by the approach that used WE in both of QE and retrieval processes. However, with regards to Topic 2 and 7, the best results were shown in BM25 retrieval model that supported by POS tagger to filter the keywords. Performance with an average of 49.28% precision (@20) and 0.4528 recall (@1000) which is increased the best results achieved during FIRE2016 by about 5.71% of precision (@20) and by 11.08% of recall (@1000) respectively.

Table 3. The top results of FIRE2016

N	Run_ID	P@20	R@1000
1	iiest_saptarashmi_bandyopadhyay_1	0.4357	0.3420
2	JU_NLP_1	0.4357	0.3295
3	dcu_fmt16_1	0.3786	0.3578
4	Our approach	0.4928	0.4528

[4] http://terrier.org/

Table 3 shows the comparison of our result of the ensemble approach and the top three results of performances of FIRE2016.

6. CONCLUSION

In this paper, we presented WE approaches in Information Extraction from microblogs. We adopted WE in both of QE and retrieval model. The results indicated that WE approach performs better than BM25. Also training a local dataset is more likely show better results rather training global datasets according to our results. In future, we would like to employ deep learning approaches such as convolutional neural network to extract more semantic information.

7. REFERENCES

[1] M.Imran, S. Elbassuoni, C. Castillo, F. Diaz, P. Meier. Extracting Information Nuggets from Disaster Related Messages in Social Media. *In: Proceeding of the 10th International ISCRAM Conference, 2013*

[2] *A.Ritter, Mausem and O.Etzioni, Open domain event extraction from Twitter. KDD'12 2012.*

[3] J.Piskorski, R.Yangerber Information Extraction: Past, Present and Future. *In: Multi source, Multilingual Information Extraction and Summarisation. 2013*

[4] L. Derczynski, A. Ritter, S. Clarke, and K. Bontcheva. 2013. "Twitter Part-of-Speech Tagging for All: Overcoming Sparse and Noisy Data". In *Proceedings of the International Conference on Recent Advances in Natural Language Processing*, ACL.

[5] Godin, F., Vandersmissen, B., De Neve, W., & Van de Walle, R. (2015). Multimedia Lab @ ACL W-NUT NER shared task: Named entity recognition for Twitter microposts using distributed word representations. *In: Workshop on Noisy User-generated Text, ACL 2015.*

[6] George A. Miller (1995). WordNet: A Lexical Database for English. *Communications of the ACM Vol. 38, No. 11: 39-41.*

[7] S.Ghosh and K.Ghosh. Overview of the FIRE 2016 Microblog track: Information Extraction from Microblogs Posted during Disaster, In Working notes of FIRE 2016 - Forum for Information Retrieval Evaluation, Kolkata, India, December 7-10, 2016, CEUR Workshop Proceeding. CEUR-WS.org, 2016

[8] J.Kaur and V.Gupta, Effective Approaches For Extraction of Keywords. In Proceedings: IJCSI International Journal of Computer Science Issues, Vol. 7, Issue 6, November 2010.

[9] D.Pal, M.Mitra and K.Datta. Improving Query Expansion Using Wordnet. CoRR, abs/1309.4938, 2013

[10] C.Carpinato and G.Romano. A survey of automatic query expansion in information retrieval. In Proceeding: ACM Comput. Surv., 44(1), 1:1-1:50, January, 2012

[11] F.Diaz, B.Mitra and N.Craswell. Query Expansion with Locally-trained Word Embeddings. In Proceeding: 54 Annual meeting of the Association for Computational Linguistics, pages 367-377. Berlin, Germany, 2016.

[12] Godin, F., Vandersmissen, B., De Neve, W., & Van de Walle, R. (2015). Multimedia Lab @ ACL W-NUT NER shared task: Named entity recognition for Twitter microposts using distributed word representations. Workshop on Noisy User- generated Text, ACL 2015.

[13] A list of cities of Nepal. Available at: https://simple.wikipedia.org/wiki/List_of_cities_in_Nepal

[14] A list of global non-government organization. Avalaible at: https://www.unodc.org/ngo/list.jsp

[15] A list of government organization of Nepal. Available at: https://en.wikipedia.org/wiki/List_of_Nepal_government_organizations

[16] S.Dasgupta, A.Kumar, D.Das, S.K.Naskar and S.Bandyopadhyay. Word Embeddings for Information Extraction from Tweets. In Working notes of FIRE 2016 - Forum for Information Retrieval Evaluation, Kolkata, India, December 7-10, 2016, CEUR Workshop Proceeding. CEUR-WS.org, 2016

[17] S.Bandyopadhyay. Correlation Distance based Information Extraction System at FIRE 2016 Microblog Track. In Working notes of FIRE 2016 - Forum for Information Retrieval Evaluation, Kolkata, India, December 7-10, 2016, CEUR Workshop Proceeding. CEUR-WS.org, 2016

Appendix A.

<num> Number: FMT1
<title> What resources were available
<desc> Identify the messages which describe the availability of some resources.
<narr> A relevant message must mention the availability of some resource like food, drinking water, shelter, clothes, blankets, human resources like volunteers, resources to build or support infrastructure, like tents, water filter, power supply and so on. Messages informing the availability of transport vehicles for assisting the resource distribution process would also be relevant. However, generalized statements without reference to any resource or messages asking for donation of money would not be relevant.

<num> Number: FMT2
<title> WHAT RESOURCES WERE REQUIRED
<desc> Identify the messages which describe the requirement or need of some resources.
<narr> A relevant message must mention the requirement / need of some resource like food, water, shelter, clothes, blankets, human resources like volunteers, resources to build or support infrastructure like tents, water filter, power supply, and so on. A message informing the requirement of transport vehicles assisting resource distribution process would also be relevant. However, generalized statements without reference to any particular resource, or messages asking for donation of money would not be relevant.

<num> Number: FMT3
<title> WHAT MEDICAL RESOURCES WERE AVAILABLE
<desc> Identify the messages which give some information about availability of medicines and other medical resources.
<narr> A relevant message must mention the availability of some medical resource like medicines, medical equipments, blood, supplementary food items (e.g., milk for infants), human resources like doctors / staff and resources to build or support medical infrastructure like tents, water filter, power supply, ambulance, etc. Generalized statements without reference to medical resources would not be relevant.

<num> Number: FMT4
<title> WHAT MEDICAL RESOURCES WERE REQUIRED
<desc> Identify the messages which describe the requirement of some medicine or other medical resources.
<narr> A relevant message must mention the requirement of some medical resource like medicines, medical equipments, supplementary food items, blood, human resources like doctors / staff and resources to build or support medical infrastructure like tents, water filter, power supply, ambulance, etc. Generalized statements without reference to medical resources would not be relevant.

<num> Number: FMT5
<title> WHAT WERE THE REQUIREMENTS / AVAILABILITY OF RESOURCES AT SPECIFIC LOCATIONS
<desc> Identify the messages which describe the requirement or availability of resources at some particular geographical location.
<narr> A relevant message must mention both the requirement or availability of some resource,(e.g., human resources like volunteers / medical staff, food, water, shelter, medical resources, tents, power supply) as well as a particular geographical location. Messages containing only the requirement / availability of some resource, without mentioning a geographical location would not be relevant.

<num> Number: FMT6
<title> WHAT WERE THE ACTIVITIES OF VARIOUS NGOs / GOVERNMENT ORGANIZATIONS
<desc> Identify the messages which describe on-ground activities of different NGOs and Government organizations.
<narr> Narrative:
A relevant message must contain information about relief-related activities of different NGOs and Government organizations in rescue and relief operation. Messages that contain information about the volunteers visiting different geographical locations would also be relevant. However, messages that do not contain the name of any NGO / Government organization would not be relevant.

<num> Number: FMT7
<title> WHAT INFRASTRUCTURE DAMAGE AND RESTORATION WERE BEING REPORTED
<desc> Identify the messages which contain information related to infrastructure damage or restoration.
<narr> A relevant message must mention the damage or restoration of some specific infrastructure resources, such as structures (e.g., dams, houses, mobile tower), communication infrastructure (e.g., roads, runways, railway), electricity, mobile or Internet connectivity, etc. Generalized statements without reference to infrastructure resources would not be relevant.

Os desafios e contribuições de Big Data para a consolidação das Smart Cities

Vanderley Gondim[1]

[1]Universidade de Evora, Évora, Portugal
d11194@alunos.uevora.pt

Resumo. As cidades atraem cada vez mais pessoas e o ambiente urbano será aquele em que a maioria dos seres humano viverão nas próximas décadas. Para responder às necessidades e exigências dos habitantes, as cidades elevaram o seu nível de sofisticação e complexidade: na distribuição de energia, abastecimento de água, rede de transporte, recolhimento de resíduos e redes de comunicação, que hoje são serviços essenciais para o funcionamento das cidades. A crescente complexidade associada ao desconhecimento sobre o tamanho do ambientes urbanos que concentram milhões de pessoas e o volume de dados que geram diariamente, exigem uma revisão dos conceitos que têm vindo a ser utilizados nos dias de hoje. A utilização de Big data no âmbito das *Smart Cities* tem provocado uma corrida pela modernizaçao urbana e utilização inteligente e eficiente da informação. Este artigo apresenta os desafios e contribuições de *Big Data* no processo de consolidação das *Smart Cities*. Ao final, verificou-se que ainda há oportunidade de pesquisas que envolvam a temática do artigo que possam evidenciar as boas práticas, as iniciativas, experiências, oportunidades e o protagonismo cidadão nesta vasta área de intervenção.

Palavras-chave: Cidades Inteligentes, Big Data, Desafios, Contribuições.

1 Introdução

Nos últimos anos, governantes de todo o mundo têm demonstrado preocupação com a crescente migração da população rural para a área urbana. De acordo com relatório da ONU [1], em 2014, 54% da população mundial residia em áreas urbanas. Em 1950, 30% da população mundial era urbana, e em 2050, 66% do total. Chuan Tao Yin et al. [2] afirmam que o processo de urbanização das cidades cria novos desafios e problemas tanto ambientais como ecológicos e também problemas de desordem pública, com o crescimento populacional e maximização de uso dos recursos naturais. O aumento da população urbana também pode afetar as áreas econômica, social, educacional, de segurança e transporte. O conceito de *Smart Cities* (Cidades Inteligentes) [3], oferece oportunidades para enfrentar grandes desafios, resolver problemas urbanos e proporcionar aos cidadãos uma melhor qualidade de vida [2]. Cidades em processo de transformação

em *Smart Cities* estão surgindo, mudando gradativamente a relação com os cidadãos e influenciando a sua forma de pensar e agir. Diante desses desafios, várias tecnologias surgem como vetores dessas mudanças, dentre elas, *Big Data* [4]. As cidades estão acumulando dados mais rapidamente do que conseguem utilizar. Desta forma, as *Smart Cities* estão aproveitando esta tecnologia para tornar esses dados acessíveis e transparentes à população. Através de análises apuradas, retiram informações elaboradas para planejar ,investir e fazer políticas de forma consciente e participativa.

Este trabalho pretende apresentar os desafios e possíveis contribuições de *Big Data* na consolidação das *Smart Cities.*

2 Estado da Arte

Contribuições acadêmicas no campo de *Big Data* e *Smart Cities* vem evoluindo desde os últimos anos. A área de pesquisa que relaciona *Big Data* e *Smart Cities* continua a ganhar força, tanto acadêmicos como profissionais tem buscado maneiras inovadoras de aliar técnicas de *Big Data* no contexto das *Smart Cities*. Nuaimi et al. [5] analisam as aplicações Big Data para apoiar Smart Cities, discute e compara diferentes definições de Smart *Cities* e *Big Data* e explora as oportunidades, desafios e benefícios da incorporaração de *Big Data* às *Smart Cities*. Além disso, tenta identificar os requisitos que suportam a implementação de *Big Data* para serviços de *Smart Cities*. O estudo revela ainda que várias oportunidades estão disponíveis para a utilização de *Big Data* em *Smart Cities*; No entanto, ainda há muitos problemas e desafios a serem abordados para conseguir uma melhor utilização desta tecnologia.

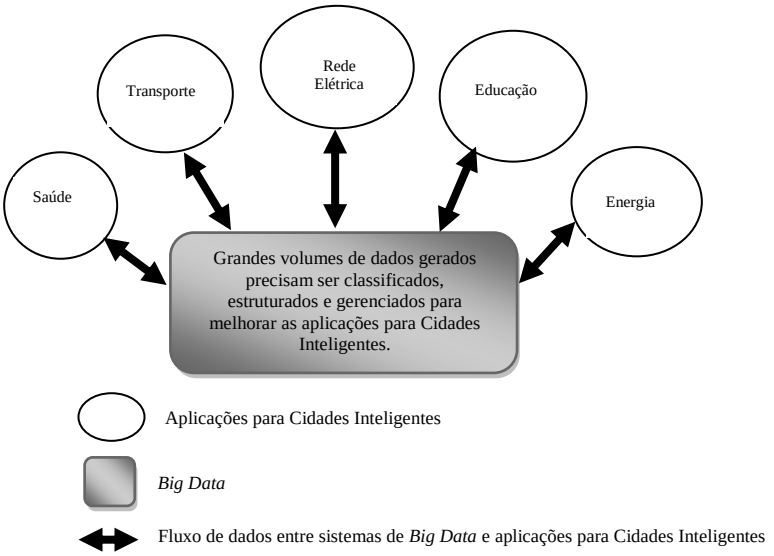


Fig 1. Relacionamento entre Cidades Inteligentes e *Big Data* (Fonte: Al Nuaimi *et al*, p. 4)**.**

A Fig 1 mostra o emprego de *Big Data* em *Smart Cities*. As aplicações de cidades inteligentes geram enormes quantidades de dados, enquanto *Big Data* utiliza estes dados para fornecer informações para melhorar as aplicações das *Smart Cities* [5]. Outra proposta que relaciona as duas temáticas é apresentada por [6]. O autor apresenta uma arquitetura *Big Data* para *Smart Cities*, intitulada BASIS, mostrando que os resultados obtidos revelam capacidade adequada para armazenar, processar, analisar e disponibilizar *Big Data* no contexto de *Smart Cities*, através de um caso de demonstração. Já [7] desenvolveu uma proposta de modelo conceitual para *Smart Cities*, que utiliza *Big Data* e dados abertos como fonte de dados. Numa outra abordagem de *Big Data*, a mobilidade urbana, um dos maiores problemas das grandes cidades, é discutida em [8] [9] [10] [11] [12].

Em um estudo de caso executado no Portal da Transparência do Governo Federal do Brasil, [13] propõem a aplicação de técnicas de programação paralela baseadas no paradigma de programação *mapreduce* para fazer a identificação de um conjunto pré-determinado de produtos comprados pela Administração Pública, além de propor uma forma de consolidação dessas informações de maneira que permita a fácil visualização de disparidades encontradas no grande volume de dados apresentados. No campo da educação, Uma contribuição para o uso de volumes de dados no campo das *Smart Cities* é vista em [14]. Os autores propõem uma abordagem para o ensino da alfabetização de dados no contexto de tarefas de inovação urbana, utilizando uma ideia de Jogos de Dados Urbanos. Uma das áreas da sociedade com maior necessidade de intervenção, a área social é abordada em [15] através de projeto de uso de *Big Data* para prever e prevenir propriedades residenciais vagas. Da mesma forma, um armazém de *Big Data* foi construído para fornecer uma única fonte de informação sobre os cidadãos Organizações e agências governamentais para que possam fornecer melhores serviços.

A utilização de sensores domésticos inteligentes, redes veiculares, sensores de tempo e água, sensores de estacionamento inteligentes e objetos de vigilância para obtenção de informações, têm trazido resultados satisfatórios e promovido qualidade de vida em algumas regiões. Empregando *Big Data Analytics* [16], foi possível analisar o uso desses sensores para facilitar o planejamento urbano, promover serviços de eco-turismo e atração de turistas [17] [18] [19].

3 Smart Cities

As cidades são a chave para o crescimento econômico dos territórios. e é nessa perspectiva que os problemas surgem. Para abordar com sucesso os problemas urbanos que surgem, a análise e a gestão de informação relevante dos cidadãos serão necessárias. Para recolher, integrar e gerir esta informação, é preciso desenvolver diferentes instrumentos, já que a tecnologia de inteligência traducional pode ser insuficiente. Atualmente, muito se discute o crescimento exponencial das *Smart Cities*, mas a sua definição ainda suscita discussões das diversas correntes ligadas ás áreas de meio-ambiente e sustentabilidade, Transportes e mobilidade, saúde, segurança pública, educação, recursos naturais e energia, infraestrutura e governança.

Após análise de dezenas de artigos com definições para Smart Cities em diversas áreas, [20] definiu *Smart Cities* como um sistema que melhora o capital humano e social com sabedoria utilizando e interagindo com recursos naturais e econômicos através de soluções baseadas em tecnologia e inovação para abordar questões públicas e eficientemente alcançar o desenvolvimento sustentável e uma alta qualidade de vida com base em múltiplas parcerias com os municípios. As cidades necessitam envolver e mobilizar as pessoas e as empresas, colaborando ativamente com informação especializada - utilizando as TIC - orientada para o cidadão e para toda a cadeia de intervenientes no setor público. Para Renata Dameri [3], as *Smart Cities* se espalharam pelo mundo e estão sendo direcionadas para espaços urbanos mais inteligentes, fazendo uso de tecnologia de ponta para a resolução de problemas sociais, ambientais, mobilidade e crescimento populacional.

Uma *Smart City* é um local onde se desenvolvem continuamente projetos que agregam valor ao cidadão, melhorando a sua qualidade de vida com o menor custo possível usando a tecnologia e inovação. Iniciativas em *Smart Cities* devem gerar emprego e riqueza nas cidades através da inteligência urbana e da promoção de oportuniddes de negócio, novos mercados, produtos, serviços e da melhoria da competitividade empresarial. Trata-se de agrupar empresas que operam em setores distintos, de forma que possam desenvolver projetos que tragam valor aos cidadãos e, ao mesmo tempo, criem empregos, riqueza, obtenham benefício econômico sustentável. o nível de vida.

4 Big Data

Desde o início do uso das Tecnologias da Informação e Comunicação (TIC) na administração pública, jamais ouviu-se falar na união entre governantes e grandes empresas - IBM, Cisco e Siemens, entre outras - em torno de uma mesma tecnologia [21]. O fenômeno *Big Data* surge com o propósito de colocar ordem no caos de informações geradas todos os dias utilizando dados que já existem mas que ainda não foram aproveitados, despertando interesse em governantes e grande empresas.
Existe na literatura atual várias definições para *Big Data*. Hashem et al. [22] afirmam que Big Data é o conjunto de métodos e tecnologias em que novas formas são integradas para revelar valores ocultos em conjuntos de dados diversos, complexos e de alto volume. Sob o aumento explosivo de dados globais, o termo *Big Data* é usado principalmente para descrever enormes conjuntos de dados. Em comparação com os conjuntos de dados tradicionais, *Big Data* normalmente inclui massas de dados não estruturados que precisam de mais análise em tempo real. Além disso, também traz novas oportunidades para descobrir novos valores, ajuda a obter uma compreensão aprofundada dos valores ocultos e também incorre em novos desafios, como, por exemplo, como organizar e gerenciar esses conjuntos de dados com eficiência [8].

4.1 Características

Segundo Furht e Villanustre [4], existem 3 características associadas a *Big Data*:

- **Volume** refere-se ao tamanho dos dados de *Terabytes* (TB) para *Petabytes* (PB) e Incluindo estruturas de dados, incluindo registros, transações, arquivos e tabelas. Espera-se que os volumes de dados cresçam 50 vezes até 2020.
- **Velocidade** refere-se a maneiras de transferir grandes dados, incluindo batch, near time, real Tempo, e córregos. A velocidade também inclui as características de tempo e latência do tratamento de dados. Os dados podem ser analisados, processados, armazenados e gerenciados em uma taxa rápida, ou com um intervalo de tempo entre os eventos.
- **Variedade** de grandes dados refere-se a diferentes formatos de dados, incluindo dados estruturados, não estruturados, semi-estruturados e a combinação destes. O formato de dados pode estar nas formas de documentos, e-mails, mensagens de texto, áudio, imagens, vídeo, dados gráficos e outros.

Além dessas três características principais de *Big Data*, Sharma e Mangat [23] acrescentam mais duas: Valor e Veracidade.

- **Valor** refere-se a benefícios ou valor obtido pelo usuário à partir de Big Data.
- **Veracidade** refere-se à qualidade de *Big Data*.

4.2 Desafios

Como toda tecnologia, *Big Data* ainda enfrenta alguns desafios, especificamente no que diz respeito ao design, desenvolvimento e implantação das Smart Cities [5]. Em [24] [25] [26], são apresentados desafios inerentes a aplicação de *Big Data*:

- **Privacidade e Segurança** - Um dos maiores desafios de *Big Data*, a segurança dos dados coletados e tratados, pode ser comprometida afetando diretamente cidadãos e organizações.
- **Acesso a dados e compartilhamento de informações** - Informações necessitam preencher requisitos de qualidade - precisa, rápida e relevante - para auxiliar na tomada de decisões e o compartilhamento de informações sem critérios pode interferir nos negócios e trazer algum tipo de transtorno aos cidadãos.
- **Questões de armazenamento e processamento** - O armazenamento de informações por si só torna-se irrelevante quando se fala em *Big Data*. Ainda que os tradicionais meios de armazenamento estejam evoluindo, a utilização de *Cloud Computing* [27] tem crescido largamente, mesmo trazendo dificuldades relacionadas ao processamento e velocidade de acesso.
- **Desafios analíticos** - A quantidade massiva de dados pode revelar alguns problemas, como a necessidade ou não de todos os dados serem armazenados e analisados, quais são realmente importantes e como podem ser aproveitados da melhor forma, sendo eles estruturados, semi-estruturados ou não estruturados.
- **Requisitos de habilidades** - Por ser recente, a adoção de *Big Data* tem demandado profissionais qualificados com capacidades de investigação, analíticas, interpretativas e analíticas.
- **Desafios técnicos** - Aspectos relacionados a tolerância a falhas, escalabilidade, qualidade dos dados e dados heterogêneos configuram-se como essenciais na utilização de *Big Data.*
- **Privacidade** - A privacidade é uma questão amplamente discutida em qualquer domínio. O vazamento de informações, por exemplo, tem colocado em risco a reputação das organização. Ações na justiça são movidas por pessoas que têm a sua intimidade violada, nações são surpreendidas com divulgação de informações

estratégicas e ainda a venda de informações pessoais e perfis de consumo para ações de marketing não autorizadas.

- **Dados incompletos e variáveis** - A qualidade dos resultados de análises de informações dependem basicamente do tipo de informação que se tem em mãos. Informações não estruturadas podem acarretar resultados inconsistentes ou equivocadas. da mesma maneira, dados com tipologias diferentes como gráficos, textos ou imagens quando utilizados em conjuntos podem trazer dificuldades para obtenção de resultados inteligíveis pelos sistemas.

5 Conclusões

O conhecimento, a informação e a gestão de grandes volumes de informação são a alavanca para os novos desafios das *Smart Cities*. Nos últimos tempos, o desafio tem sido combinar a ambição da administração pública com soluções tecnológicas e sistemas inteligentes com o propósito de gerar eficiência em todos os processos de gestão das cidades, sempre pensando no cidadão. Um desafio obriga as cidades a se reinventar, tornando-se mais competitivas e com serviços de excelência para pessoas cada vez mais exigentes. Desta forma, tratamos neste artigo dos desafios enfrentados pelas *Smart Cities* na utilização de *Big Data* e as contribuições de seu uso. Analisando os artigos que tratam da aplicação de *Big Data* em *Smart Cities*, percebe-se que há um desequilíbrio na quantidade de produções científicas, especificamente nas áreas de educação e social. As pesquisas sobre os temas ainda carecem de maior aprofundamento e os estudos de caso poderiam ser mais explorados.

Assim, as cidades vão precisar, cada vez mais, encontrar novas formas de envolver os seus cidadãos nas tomadas de decisão. A necessidade de identificar questões ligadas a geração, armazenamento e tratamento dos dados estarão sempre na pauta de discussão.

Referências

1. UN. United Nations, Department of Economic and Social Affairs, Population Division. *World Urbanization Prospects: The 2014 Revision,* (ST/ESA/SER.A/366). Highlights.pdf. 2015.
2. Yin C T, Xiong Z, Chen H, et al. *A literature survey on smart cities*. Sci China Inf Sci: 100102(18), doi: 10.1007/s11432-015-5397-4. 2015.
3. Dameri, R. P. *Searching for Smart City definition: a comprehensive proposal*. International Journal of Computers & Technology, Vol. 11, nº 05, pp. 2544-2551. Genova, Italy. 2013.
4. Furht, B. e Villanustre, F. *Big Data Technologies and Applications*. Springer International Publishing Switzerland. DOI 10.1007/978-3-319-44550-2_1. 2016.

5. Nuaimi, E. A., Neyadi, H. A., Mohamed N. e Al-Jaroodi, J. *Applications of big data to smart cities.* Journal of Internet Services and Applications. Springer Open Journal. DOI 10.1186/s13174-015-0041-5. 2015.
6. Costa, C. *BASIS:* Uma Arquitetura de *Big Data* para *Smart Cities*. Dissertação de Mestrado. Universidade do Minho, Braga. 2015.
7. Klein, V. B. Uma proposta de Modelo Conceitual para uso de Big Data e Open Data para Smart Cities. Dissertação de Mestrado. UFSC, Florianópolis, SC. 2015.
8. Chen, M., Mao, S. e Liu, S. *Big Data: A Survey*. Mobile Netw Appl 19:171–209. Springer Science+Business Media New York. DOI 10.1007/s11036-013-0489-0. 2014.
9. Bravo, Y., Ferrer, J., Luque, G. e Alba, E. *Smart Mobility by Optimizing the Traffic Lights: A New Tool for Traffic Control Centers*. Smart-CT 2016, LNCS 9704, pp. 147–156. Springer International Publishing Switzerland. DOI: 10.1007/978-3-319-39595-115. 2016.
10. Kemp, G., Vargas-Solar, G., Silva, C. F., Ghodous, P., Collet, C. et al. Towards Cloud big data services for intelligent transport systems. concurrent engineering, July 2015, Delft, Netherlands. 2015.
11. Mehmood, R. e Graham, G. Big data logistics: a health-care transport capacity sharing model. Procedia Computer Science, Vol. 64, pp. 1107-1114. 2015.
12. Niu, X., Zhu, Y., Cao, Q., Zhang, X., Xie, W. e Zheng, K.. An online-traffic-prediction based route finding mechanism for smart city. International Journal of Distributed Sensor Networks, Vol. 2015, p. 16. 2015.
13. Paiva, E. e Revoredo, K.. Big Data and Transparency: Using MapReduce functions to increase Public Expenditure transparency. *Proceedings of the SBSI 2016*, May 17–20. Florianópolis, Santa Catarina, Brazil. 2016.
14. Wolff, A., Kortuem, G. e Cavero, J. Urban data games: creating smart citizens for smart cities, *IEEE - Proceedings of the 15th International Conference on Advanced Learning Technologies (ICALT)*, Hualien, pp. 164-165. 2015.
15. Kim, G.H., Trimi, S. and Chung, J.H. *Big-data applications in the government sector*, Communications of the ACM, Vol. 57 No. 3, pp. 78-85. 2014.
16. Thakuriah, P. Tilahun, N. e Zellner, M. *Seeing Cities Through Big Data*, Springer Geography, Switzerland. DOI 10.1007/978-3-319-40902-3_29. 2017.
17. Rathore, M.M., Paul, A., Ahmad, A. e Rho, S. *Urban planning and building smart cities based on the internet of things using big data analytics*, nº 101, pp. 63-80. Computer Networks. DOI: dx.doi.org/10.1016/j.comnet.2015.12.023. 2016.
18. Toppeta, D. 2010. *The smart city vision: how innovation and ICT can build smart, 'liveable', sustainable cities, Innovation Knowledge Foundation* (Think! Report 005/2010), disponível em: www.inta-aivn.org/images/cc/Urbanism/background%20documents/Toppeta_Report_005_2010.pdf. Acesso: 05 de Janeiro de 2017.
19. Sobolevsky, S., Bojic, I., Belyi, A., Sitko, I., Hawelka, B., Murillo Arias, J. e Ratti, C. Scaling of city attractiveness for foreign visitors through big data of human economical and social media activity. *Proceedings of the International Congress on Big Data* (BigData Congress), New York, NY. 2015.

20. Fernandez-Anez, V. *Stakeholders Approach to Smart Cities: A Survey on Smart City Definitions*. Smart-CT 2016, LNCS 9704, pp. 157–167. Springer International Publishing Switzerland. 2016.
21. Townsend, A. M. Smart Cities: big data, civic hackers, and the quest for a new utopia. W.W. Norton & Company. New York, NY. 2013.
22. Hashem, I. A. T., Yaqoob, I., Anuar, N. B., Mokhtar, S., Gani, A., e Khan, S. U. *The rise of "big data" on cloud computing: Review and open research issues. Information Systems*, 47, 98-115. 2015.
23. Sharma, S. e Mangat, V. *Technology and trends to handle big data: a survey*. In: fifth international conference on advanced computing and communication technologies. p. 266–71. 2015.
24. Menon, S. P. e Hegde, N. P. *A Survey of Tools and Applications in Big Data*. IEEE 9th international Conference on intelligent Systems and Control (JSC0). 2015.
25. Avita, K., Wazid, M. e Goudar, R. H. *Big data: Issues, challenges, tools and Good practices*. In Contemporary Computing (lC3), 20 13 Sixth International Conference on, pp. 404-409. IEEE. 2013.
26. Misra, R., Panda, B. e Tiwary, M. *Big data and ICT applications – A study*. ICTCS '16. ACM. DOI: http://dx.doi.org/10.1145/2905055.2905099. 2016.
27. Hung P.C.K. (ed.). Big Data Applications and Use Cases, International Series on Computer Entertainment and Media Technology, Springer International Publishing Switzerland. DOI 10.1007/978-3-319-30146-4_1. 2016.

A Survey of Automatic Personality Recognition in Online Social Networks

Ganchimeg Lkhagvasuren

Universidade de Évora
ganchimeg@seas.num.edu.mn

Abstract. Recognising personality profiles in user-generated content on online social networks is very valuable for applications that rely on personalisation, such as personalised advertising and recommender systems. This survey tries to explain current methods used for recognising personalities and compare them. In this work, we survey 10 approaches which most of them were participated to evaluation campaigns using Facebook, Twitter and Youtube data.

Keywords: personality trait, personality recognition, online social network, computational personality recognition, author profiling task, automatic personality recognition, social network, Facebook profile feature, social medium, status update, youtube dataset.

1 Introduction

It is undeniable that we live online with social networks. With the Internet, every minute a huge number of textual data is added mostly by individual users. In particular, there were about 1.04 billion daily active users on average for March 2015 in Facebook [1]. This growing big amount of data has made a challenge that can be played by many researchers on intensive interesting explorations. One of those researches is a detection of author personality profile based on the text that she or he has written [2]. Many recent studies, such as [4] [14] [16] [17] and so on, have stated that the personality of individual users correlate their style of writing on post, tweet, comment and so on. To determine the personality profile, Big Five personality traits [3] are used widely in the bulk of studies.

The majority of the researchers used various procedures and evaluation measures in this area, consequently, describing the state-of-the-art is not easy regarding the classification effectiveness [15].

In this paper, we survey approaches of personality detection evaluated on datasets of social network sites, such as Youtube dataset, Facebook posts and Twitter tweets, and try to analyse them.

The paper is organised as follow: In section 2 we give an overview of the related work done. In section 3 we provide common knowledge of the Big five, information of available corpora and brief description of features used in personality estimation. In section 4 we introduce all approaches by categorising 3 sections based on the type of datasets. In section 5 we try to compare all approaches and finally in section 6 our work is concluded.

Not much work has been done with survey of Automatic personality recognition in online social networks. Studies measuring personality are often confined because of not much volunteers are participated in the studies since people avoid reporting their personal behaviours and preferences under certain situations [14].

Vinciarelli et al [6] aimed to provide a solid knowledge base about the state-of-the-art and make a survey of not only Automatic personality recognition (APR, the recognition of the personality of an individual), but also Automatic personality perception (APP, the prediction of the personality others attribute to a given individual) and Automatic personality synthesis (APS, the generation of artificial personalities through embodied agents). As the authors stated, it is the first survey in Personalty computing. In its subsection, APR on Social Media, the authors introduced a synopsis of data, approaches and results of 12 works has been done before in APR. The samples and features of these works are diverse. For example, one of these works [7] predicts personality traits using some features such as the profile info, the density of users egocentric networks and Linguistic Inquiry Word Count (LIWC) [18]. Also [28] consider 473 posts on FriendFeed based on some LIWC categories while the work in [36] labels 10000 users of *Livejournal* (a blogging site). The performance of these works was reported in terms of F-measure, Root mean square error, Mean absolute error and Accuracy.

We will present works that so far has been done in online social networks from when the survey [6] released in our analysis.

2 Background

This section outlines the concepts of Big Five with particular emphasis on its every traits, the introduction of available corpora for a research of detecting author personality and brief overview of techniques so far have been used to APR.

2.1 The Big Five

In this part, although there are multiple ways [19][20][21] to classify traits, we introduce Big Five factor model of personality since all the works surveyed in this paper have selected it. The idea of Big Five is that an individual can be identified with

five scores that correspond to five main personality traits [9]. Table 1 collates a brief explanation of each trait along with descriptive terms that are commonly associated with them.

Openness *involves six facets including active imagination (fantasy), aesthetic sensitivity, attentiveness to inner feelings, preference for variety, and intellectual curiosity [22] [26].*

Conscientiousness *is the personality trait of being thorough, careful, or vigilant. This type of people are efficient and organised as opposed to easy-going and disorderly [23].*

Extraversion *tends to be manifested in outgoing, talkative, energetic behaviour, whereas introversion is manifested in more reserved and solitary behaviour [24].*

Agreeableness *is a personality trait manifesting itself in individual behavioural characteristics that are perceived as kind, sympathetic, cooperative, warm and considerate [27].*

Neuroticism *is a fundamental personality trait in the study of psychology characterised by anxiety, fear, moodiness, worry, envy, frustration, jealousy, and loneliness* [25].

Table 1. The big five personality dimensions

Personality traits	**High scores**	**Low scores**
Openness	Imaginative	Conventional
Conscientiousness	Organized	Spontaneous
Extraversion	Outgoing	Solitary
Agreeableness	Trusting	Competitive
Neuroticism	Prone to stress to worry	Emotionally stable

Extensive studies have used the big five as the current definitive model of personality, however, Vinciarelli et al [10] assumes personality scores are still inadequate. Their consideration is "*There are many cases where the gap between data and traits is so wide, machine intelligence methodologies (signal processing, machine learning, natural language processing, etc.) cannot achieve satisfactory results without converting personality scores into binary variables, easier to deal with*." Also the authors point out "*We need more personality in Personality Computing*.".

2.2 Available corpora

In recent a decade, besides many attempts have tried to define user personalty, evolution campaigns [12-13] [29] have contributed considerably to evaluate different approaches on a common benchmark and the making of publicly datasets which are valuable resources to researchers.

One of these campaign, The workshop on Computational personality, released a corpora which consists 9917 status updates of 250 users from *MyPersonality project*[1] [13] and a corpora which consists transcript of youtube blogs (youtube dataset)[2] and mobile phone interactions(mobile dataset) [12] in 2013 and in 2014 respectively.

The another evaluation champaign [29], Author profiling task PAN at CLEF 2015, provided participants with a training data set that consists of Twitter tweets[3] in English, Spanish, Italian and Dutch. Each dataset of different languages has labels of gender (male, female), age (18-24, 25-34, 35-49, 50-xx) and five personality traits values between -0.5 and 0.5. Age data for Italian and Dutch languages are not available.

2.3 Features

Most approaches used two basic types of features that can be used for authorship profiling: content-based features and style based features [34]. In this section, we introduce briefly some features mostly used in approaches that is surveying in our paper.

Linguistic Inquiry and Word Count (LIWC)

LIWC[4] is a text analysis tool that provides a broad range of social and physiological insights. This tool includes the main text analysis module along with a group of built-in dictionaries. After the processing module has read and accounted for all words in a given text, it calculates the percentage of total words that match each of

---

[1] http://mypersonality.org/wiki/doku.php?id=download_databases#datasets_available_without_registration

[2] https://sites.google.com/site/wcprst/home/wcpr14#data

[3] http://www.uni-weimar.de/medien/webis/events/pan-15/pan15-web/author-profiling.html

[4] http://www.liwc.net

the dictionary categories. For example, if LIWC analysed a single speech that was 2,000 words and compared them to the built-in LIWC2015 dictionary, it might find that there were 150 pronouns and 84 positive emotion words used. It would convert these numbers to percentages, 7.5% pronouns and 4.2% positive emotion words.

Medical Research Council (MRC)

MRC[5] database of psycholinguistic categories is an online service (since version 1) and machine usable dictionary (since version 2) that can be freely utilised for the purposes of natural language processing and artificial intelligence task [35].

3 Personality Recognition in online social networks

The social sites concepts are different and its datasets of social have own specific features. It is more likely that what features to be employed is depended on what datasets to be used. Moreover, Our introducing approaches were associated with social sites mostly such as Facebook, Twitter, Youtube data. Hence, we split all approaches into 3 subsections, namely Approaches in Facebook, Approaches in Twitter and Approaches in Youtube.

3.1 Approaches in Facebook

In the workshop on computational personality recognition 2013 (WCPR13), there were 8 teams participating to the shared task. The approaches ranked down use lexical resources (e.g, the LIWC) while better ones based on words and N-grams. Let us to consider only best two works ranked by organisers. The first work [11], distinguishing it from the other work in the campaign is that the used ensemble methods based on meta learning to generalise features across genres. First they employed 2000 frequent trigrams as initial features taken from the Essays corpus, then exploited the ensemble methods, finally trained Support Vector Machines classifiers [13].

In the next work [9], the authors tried to predict personality traits based on Facebook status updates, network properties and time factors using machine learning techniques. They sampled 9917 status updates and 250 users which were collected from myPersonality project besides a corpus of 2468 essays. 81 LIWC features, 7 features related to the social network such as network size and density, 6 features

[5] http://www.websites.psychology.uwa.edu.au/school/MRCDatabase/uwa_mrc/htm

related to the time of the status, and 6 other features such as total number of statuses per user and number of capitalised words are exploited in the study. Also authors picked up 3 learning algorithms trained on these features, namely Support Vector Machine with a linear kernel (SVM), Nearest Neighbour with k=1 (kNN) and Naive Bayes (NB), and compared the performance of them.

Moreover, in order to detect personality, age and gender, a study [16] presented an *open-vocabulary* approach in which the words analysed are based on the data itself, and they extracted 700 million *words*, *phrases* and *topics* (words, phrases: a sequence of 1 to 3 words, topics: a cluster of semantically related words) from a huge number of Facebook messages of 75,000 volunteers. The authors stated that they found the *language* that correlates with personality traits. Top 5 words and phrases with best correlation strength are listed in Table 2.

Table 2. Top 5 words and phrases ranked by correlation strength in scales of personality traits.

Trait	Scales	Top 5 words and phrases ranked by correlation strength
Extraversion	Extraversion	party, cant_wait, girls, love_you, im
	Introversion	anime, xD, internet, :_3, computer
Neouroticism	Neoroticism	fucking, sick_of, fuck, I_hate, anymore
	Emotional stabilty	success, lakers, basketball, smh, beautiful_day
Agreeableness	High agreeableness	wonderful, amazing, excited, prayers, a_great
	Low agreeableness	fucking, shit, fuck, bitch. hell
Conscientiousness	High conscientiousness	to_work, ready_for, great_day, thankful, blessed
	Low conscientiousness	fucking, fuck, d:, youtube, pokemon
Openness	High Openness	universe, writing, I've, art, music
	Low openness	can't, wait, don't, ur, u

The study [14] is divided into two main sections, namely Personalty vs website preference and Personality vs Facebook profile features. The Facebook profile features were acquired from extensive user profiles that is more than 354,000 US Facebook users who had used Facebook for at least 2 years before the data was recorded. Based on multiple Facebook profile features such as the number of friends, groups joined and the number of Facebook Likes, they predicted the big five

personality using the multivariate linear regression method. Each of personality traits were predicted better with a different subset of Facebook profile features.

The study [17] attempted to find out how user's personality affects how people interact and communicate in Facebook. This study can be seen as an improved version of system for personality recognition that exploits linguistic cues and does not require supervision for evaluation in Celli et al [22]. In the dataset they used, there are 5000 post and 1100 Italian users. For about the test set, they sampled statuses of 23 Facebook users who took the Big Five personality test. The cross language features, for instance: punctuation, exclamation marks, parentheses, question marks, quotes, word reputation and average work frequency, were extracted.

In addition, unlike the other works, the study [37] presents the issue of personality and interaction style recognition from profile pictures in Facebook. They collected a number of profile pictures and labeled some of them as a gold standard. As for methodology, they employed a bag-of-visual-words to extract features from pictures. Then, several different machine learning techniques were used. Their best result implied that profile pictures are useful to identify personality recognition and interaction style treats because it is more likely the profile pictures convey a lot of information about its owners.

3.2 Approaches in Twitter

One of evolution campaigns, Author profiling task PAN 2015 - CLEF [29], is very valuable to compare different approaches of personality recognition. This evaluation lab has been organised since 2013. In 2015, besides the focus on age and gender identification, the task personality recognition was introduced with a dataset in four different language namely English, Spanish, Italian and Dutch. All approaches considered the task as a machine learning problem, in particular, a classification and regression problem to predict personality traits.

In our analysis, we present the two top ranked works of all 22 works submitted to the evaluation campaign. Both them employed Support Vector Machines (SVM).

The first work [30] aimed to use Latent Semantic Analysis (LSA) with Second Order Attributes (SOA) based on relationships among terms, documents, profiles, and sub-profiles jointly. Also the authors compared the performance of the proposal with a standard BOW and found out that the combination of LSA and SOA outperforms the BOW.

The second work [31] focused on features that are corpus dependent tags, character and POS N-grams. They extracted rules from the corpus to create dependent tags that will help the classifier to improve its performance. First they replaced mentions, urls, and hashtags using this rules. After POS and character n-grams was extracted, certain tokens were relabelled to extract extra grammatical information.

Table 3. Comparison of all approaches.

Ref.	Samples	Features	Meas.	Ext.	Agr.	Con.	Neu.	Ope.	Average
[9]	250 users and about 9917 status updates	LIWC, Social network features, time related features and Other features	F	0.62	0.53	0.54	0.56	0.61	
[11]	250 users and about 9917 status updates	2000 frequent trigrams 10 meta features	F	0.79	0.70	0.67	0.72	0.86	
[14]	9515-18720 Facebook profiles	Facebook profile features	A	0.31	0.05	0.16	0.23	0.11	
[16]	19 million Facebook status written by 136,000 volunteers	LIWC and open-vocabulary	R	0.38	0.31	0.35	0.31	0.42	
[17]	5200 posts and 1100 users	Some LIWC categories	F						62.8
[30]	152 tweet documents of author profiles.	Latent Semantic Analysis with Second Order Attributes	A	0.87	0.80	0.78	0.85	0.86	
[31]	152 tweet documents of author profiles.	POS and Character N-grams	RMSE	0.13	0.11	0.14	0.21	0.14	
[32]	404 users/video with transcripts	Lexical, LIWC, POS, Emotional	F	0.71	0.76	0.61	0.61	0.65	
[33]	404 users/video with transcripts	Gender, Audio-video, LIWC, NRC, MRC, Senti-Strength, SPLICE	RMSE	0.91	0.72	0.64	0.70	0.77	
[37]	about 100 profile pictures	BOW	F	0.61	0.66	0.73	0.60	0.73	

3.3 Approaches in Youtube

The WCPR14 [12] released two datasets: one of transcript of Youtube Vlogs and one of Mobile Phone interaction. The youtube dataset contains 404 users/videos with transcript and observed personality labels. There were two different task: an open shared task, where participants can do any kind of experiment, and a competition. They had 6 papers accepted, all of them used the youtube datasets. In our survey, we introduce two of them which were ranked at top.

The work [32] showed good results in the WCPR14 combining models for each traits. They exploited emotional and POS, a large feature space including psycholinguistic as well as audio visual features provided with the dataset. Also they studied whether it is possible that predict better a trait by identifying other traits.

The next approach [33] explored the multivariate regression techniques to make a combined prediction instead of training 5 learners separately to predict the 5

personalty traits. They tested 6 different regression models (3 variants of Target Stacking, 2 of Ensemble Regression Chains and one Random Forest) based on audio-video, textual (emotional and linguistic) features (LIWC, MRC, SPLICE, NRC, SentiStrength) extracted from the transcript of vlogs.

4 Comparison

It is not easy to compare approaches of studies in automatic personality recognition field because their datasets, and evaluation metrics they used, are different. However, Table 3 is structured in order to compare all approaches. We can see all approaches mentioned in Section 3 with their samples, features and results in this table. The measurements that they used are diverse such as accuracy (A), F-measure (F), root mean square error (RMSE) and square root of the coefficient of determination (R). The study [17] provided only the average result whereas the rest of the studies revealed the detailed achievements for every traits.

5 Conclusion

So far, recognising personality automatically has been addressed its own way due to its approaches are more likely uncorrelated and independent [6]. Hence the state-of-the-art is still in pieces: with a few exceptions, the experiments are performed over ad-hoc, proprietary data. However, some major campaigns have contributed considerably to not only the evaluation of possible approaches but also the release of some public data in automatic personality recognition task. In this study, we proposed to make a survey of some different approaches used to detect personality task. Overall, we included 10 works that was done on social network data, and we presented their features used, samples and results in this paper briefly. One study of those works used an image dataset whereas the others mainly worked on text datasets.

Preferences

1. Facebook. << Facebook newsroom>>, http://newsroom.fb.com/company-info
2. Kocher, M.: UniNE at CLEF 2015: Author Profiling - Notebook for PAN at CLEF 2015. In Linda Cappellato, Nicola Ferro, Gareth Jones, and Eric San Juan, editors, CLEF 2015 Evaluation Labs and Workshop – Working Notes Papers, 8-11 September, Toulouse, France, September 2015. CEUR-WS.org. ISSN 1613-0073.
3. R.T. Costa and R.R. McCrea. The revised neo personality inventory (neo-pi-r). The SAGE handbook of personality theory and assessment, 2:179-198, 2008.

4. Golbeck, J., Robles, C., Turner, K.: Predicting personality with social media. In: CHI'11 Extended Abstracts on Human Factors in Computing Systems. pp. 253-262. ACM (2011)
5. F.Mairesse, M.Walker. Automatic recognition of personality in conversation. In: Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers, Pages 85-88, 2006.
6. Vinciarelli, A., Mohammadi, G.: A survey of personality computing. IEEE Transactions on Affective Computing 5(3), 1-1 (2014)
7. Golbeck, J., Robles, C., Turner, K.: Predicting personality with social media. In: Proceeding of the Extended Abstract on Human Factor in Computing System. pp. 253-262. 2012.
8. Yaman, O.: Personality recognition through language and internet activities. 2011
9. Quercia, D., Lambiotte, R., Stillwell, D., Kosinski, M., Crowcroft, J.: The personality of popular Facebook users. In: Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work. pp. 955-964. ACM 2012
10. Vinciarelli, A., Mohammadi, G.: More personality in personality computing. In: Proceedings of IEEE Transactions on Affective Computing. Volume:5, Issue: 3, July - Sept. 1 2014.
11. Verhoeven, B., Daelemans, W., Smelt, T.D., Ensemble methods for personality recognition. In: Proceedings of the Workshop on Computational Personality Recognition (Shared task). Boston MA, 2013.
12. Celli, F., Lepri, B., Biel, J.I., Gatica-Perez, D., Riccardi, G., Pianesi, F.: The workshop on computational personality recognition 2014. In: Proceedings of the ACM International Conference on Multimedia. pp. 1245-1246. ACM (2014)
13.Celli, F., Pianesi, F., Stillwell, D., Kosinski, M.: Workshop on computational personality recognition (shared task). In: Proceedings of the ACM International Conference on Multimedia. 2013
14. Kosinski, M., Bachrach, Y., Kohl, P., Stillwell, D., Graepel, T.: Manifestations of user personality in website choice and behaviour on online social networks. Machine learning pp. 1-24 (2013)
15.Rangel, F., Celli, F., Rosso., P., Potthast, M., Stein, B., Daelemans, W.: Overview of the 3rd Author profiling task at PAN 2015. (2015)
16. Schwartz, H.A., Eichstaedt, J.C., Kern, M.L., Dziurzynski, L., Ramones, S.M., Agrawal, M., Shah, A., Kosinski, M., Stillwell, D., Seligman, M.E., et al.: Personality, gender, and age in the language of social media: The open-vocabulary approach. PloS one 8(9), 773–791 (2013)
17.Celli, F., Polonio, L.: Relationships between personality and interactions in facebook. In: Social Networking: Recent Trends, Emerging Issues and Future Outlook, pp. 41–54. Nova Science Publishers, Inc (2013)

18.Pennebaker, J.W, Chung C.K., Ireland, M., Gonzales, A., Booth R.J.: The development and psychometric properties of liwc2007 the university of texas at austin. LIWCNET 1:1-22. (2007)
19.Briggs Myers, I., McCauley, M.H., Quenk, N., Hammer, A.: A guide to the development and use of the Myers-Briggs Type Indicator (3r ed.). Consulting Psychologists Press. (1998)
20.Cattell, H.E.P., Mead, A.D.: The sixteen personality factor questionnaire (16PF): SAGE Knowledge. The SAGE handbook of personality theory and assessment. Retrieved May 27, 2013, from http://people.wku.edu/richard.miller/520 16PF Cattell and Mead.pdf
21.Eysenck, H.J., Eysenck, A.B.: Manual for EPQ-R. San Diag, SA: EdlTS. (1991)
22. Goldberg L. R. (1993). "The structure of phenotypic personality traits". *American Psychologist* 48 (1): 26–34.
23.In Wikipedia, https://en.wikipedia.org/wiki/Conscientiousness
24.In Wikipedia, https://en.wikipedia.org/wiki/Extraversion_and_introversion
25.In Wikipedia, https://en.wikipedia.org/wiki/Neuroticism
26.In Wikipedia, https://en.wikipedia.org/wiki/Openness_to_experience
27.In Wikipedia, https://en.wikipedia.org/wiki/Agreeableness
28. Celli, F.: Unsupervised personality recognition for social network sites. In: Proceedings of the International Conference on Digital Society , pp. 59-62, (2012)
29. Rangel, F., Celli, F., Posso, P., Potthast, M., Stein, B., Daelemans, W.: Overview of the 3rd Author profiling task at AN 2015. In working notes Papers of the CLEF 2015 Evaluation labs, CEUR Workshop Prooceedings. CLEF and CEUR-WS.org (2015)
30. Alvarez-Cortona, M.A., Lopez-Monroy, A.P., Montes-Y-Gomez, M., Vilblasenor-Pineda, L., Jair-Escalante, H.: Inaoe's participation at pan'15: Author profiling task - notebook for pan at clef 2015.
31.Gonzales-Gallardo, C.E., Montes, A., Sierra, G., Nunez, A., Adolfo, S., Ek, J.: Tweets classification using corpus dependent tags, character and pos n-grams - notebook for pan et clef 2015.
32.Alam, F., Riccardi, G.: Predicting personality traits using multimodal information. In: Proceedings of the 2014 ACM Multi Media on Workshop on Computational Personality Recognition (2014)
33. Fernando, G., Sushmita, S., Sitaraman, G., Ton, N., Cock, M.D., Davalos.: A multivariate regressio approach to personality impression recognition of vloggers. In: Proceedings of the 2014 ACM Multi Media on Workshop on Computational Personality Recognition (2014)
34. Argamon, A., Koppel, M., Pennebaker, J. Schler J.: Automatically profiling the author of an anonymous text.
35.Poria, S., Gelbukh, A., Agarwal, B., Cambria, E., Howard, N.: Common sense knowledge based personality recognition from text.

36.Nguyen T., Phung D., Adams B., Venkatesh S.: Towards discovery of influence and personality traits through social link prediction. In: Proceedings of the International AAAI Conference on Weblog and Social Media, 2011, pp. 566-569.
37.F.Celli, E.Bruni, B.Lepri. Automatic Personality and Interaction Style Recognition from Facebook Profile Pictures. In: Proceedings of ACMMM 2014, Orlando, Florida.

Building arguments through clustering technique

(*An extended version of this work was submitted to the ICAIL 2017 Conference*)

Prakash Poudyal

Department of Computer Science, University of Evora,
Evora, Portugal
prakashpoudyal@gmail.com

Abstract. The paper is about the clustering technique applied to cluster the argumentative sentences to form an argument. One potential problem in identifying argumentative sentences among text is that an argumentative sentence from one specific argument can also be part of another argument. To address this issue, the fuzzy clustering technique is applied such that these occurrences are identified. FCA generates the membership values ranges in 0 to 1 to the respective cluster depending upon the kinds of features that are being used. Three kinds of features are selected: N-gram, Word2Vec, Sentence position. The corpus is sharpened by using preprocessing techniques such as co-reference resolution and Wordnet to make the text more suited to feature extraction. During the process of analysis, two algorithms were developed. "Distribution of Sentence to the respective Cluster Algorithm (DSCA)" and "Appropriate Cluster Identification Algorithm (ACIA)". Overall, the result that we achieved was satisfactory. For the evaluation, Case-Laws from the European Court of Human Rights (ECHR) annotated by Mochales-Palau and Moens [7] were selected.

Keywords: argument, Fuzzy Clustering, natural language analysis

1 Introduction

Argumentation, as a branch of philosophy, plays a cardinal role in encouraging fundamental insight into the issue at hand. It is synonymous with dialectic in the sense that it assists in bringing forth arguments which may affirm or negate the particulars of an issue. Thus argumentation is an important component of human communication, its effect is increased in the current era. With the continued advancement of technology and appearance of arguments in numerous news portals, blogs, fora, and social media is growing exponentially. One of the most effective domains of the argument is in the legal corpus. In the document, we found the components of the arguments are in various structure; sequential, scattered and ambiguous nature.

In this paper, we propose the model that accumulate the components of arguments to form an argument. The task is quite challenging because components

of one argument (premise or conclusion) can also be involved in another argument as we mention above (ambiguous nature). After extracting the features associated with each text, the fuzzy clustering algorithm is applied to get a membership value ranging from 0 to 1 for every sentence. To obtain the composition of each cluster, we developed an algorithm called "Distribution of Sentence to the respective Cluster (DSCA)". Furthermore, and for the evaluation process, we need an algorithm to match our system's output with the golden standard data set. To handle this problem, we developed an algorithm "Appropriate Cluster Identification Algorithm (ACIA)" which helps to map each cluster of the newly developed system to the closest cluster of the golden standard data sets. For the evaluation, Case-Laws from the European Court of Human Rights (ECHR) annotated by Mochales-Palau and Moens [7] were selected. Details of the corpus were described in a previous publication [9].

In the argument mining field, the author has been unable to identify any research into the use of clustering techniques to identify and group argumentative sentences into arguments. However, similar problems are solved by other researchers by means of the boundary detection technique. Mochales and Moens [6] proposed Context-free grammars (CFG) to detect the argument structure and obtained a 60% accuracy. Stab and Gurevych [8] proposed an approach to identify the relation (support and attack) between the components of arguments. Likewise, Lawrence et al. [4] performed a manual analysis as well as an automatic analysis to deal with the boundaries of the argument. We propose to solve the problem through the clustering techniques. However, Clustering techniques were not considered as an appropriate technique for the information retrieval in the 1980s. The main reason was the computational complexity associated with this process as well as a lack of accuracy. To solve such problem, in 1988 Cutting et al. [1] proposed a new approach to cluster the document called "Scatter/Gather". Likewise, Huang et al. [3] compared and analyzed the effectiveness of the distance function and similarity measure in partial clustering of text documents. The author evaluates five measures with empirical experiments: Euclidean distance, Cosine Similarity, Jaccard coefficient, Pearson correlation coefficient, and averaged Kullback-Leibler divergence.

The rest of the paper is organized as follows: Section 2 deals with the concepts and tools that were used while conducting the evaluation. In Section 3 we describe the proposed architecture. This section includes the description of features and newly developed DSCA and ACIA algorithms. Section 4 evaluates the performance carried out by all the experiments. Finally, in Section 5 we address the conclusion and future work.

2 Concepts and Tools

In this section, we described the fuzzy clustering algorithm and other various tools that are used for the experiment.

2.1 Weka

Weka [2] is an acronym for "Waikato Environment for Knowledge Analysis". It is an open source data mining software under the General Public License. It consists of 49 data preprocessing tools, 76 classification/regression algorithms, 8 clustering algorithms, 15 subset evaluators, 10 search algorithms for feature selection, 3 algorithms for finding association rules. For our experiments, we used "String to Word Vector", TF-IDF and other various tools to extract the features from the dataset.

2.2 Evaluation Setup

Precision, Recall [12] [13] are the measurement tools that are being used for evaluating the performance of information retrieval system. These tools can be applied in the analysis of clustering technique as well. Suppose cluster α of golden standard data sets consists of N_α number of the sentence. Similarly, cluster$_\beta$ of the predicted system consists of N_β number of sentences. Precision ($P_{\alpha\beta}$) and Recall ($P_{\alpha\beta}$) is defined in equation 1 and 2. Similarly, its weighted is expressed in 3 and 4 and f-measure is in 5. The equations are:

$$P_{\alpha\beta} = \frac{n_{\alpha\beta}}{N_\beta} \tag{1}$$

$$R_{\alpha\beta} = \frac{n_{\alpha\beta}}{N_\alpha} \tag{2}$$

$$P = \frac{\sum_{\beta=1}^{c} N_\beta P_{\alpha\beta}}{\sum_{\beta=1}^{c} N_\beta} \tag{3}$$

$$R = \frac{\sum_{\beta=1}^{c} N_\beta R_{\alpha\beta}}{\sum_{\beta=1}^{c} N_\beta} \tag{4}$$

$$F = 2 \cdot \frac{precision \cdot recall}{precision + recall} \tag{5}$$

Here, $n_{\alpha\beta}$ is the total number of sentence matched in between cluster α and cluster β. Overall performance is measured by calculating the weight average of the individual precision (P) and recall (R) which are expressed in formula 3 and 4.

3 Proposed Architecture

In this paper, we aim to propose a system to cluster argumentative sentences and, this way, to identify arguments. There are several phases first: text processing and feature extraction; second: developing new algorithms and third (and final stage): system evaluation.

3.1 Textual Analyzer

Before extracting features, we perform some preprocessing techniques, such as co-reference resolution, and use Wordnet to make the text more suited for feature extraction. In linguistics, different terminologies are used to express the same meaning. For example, "Hari likes to eat Nepalese food because he likes spicy food." In this example, proper noun "Hari" and the pronoun "he" refers to the same person, but technically, these two words are different. In this case Co-reference resolution technique is applied to replace the word "he " with the "Hari " or vice-versa. Similarly, Wordnet [5] technique helps to increase the accuracy result by appending the synonym of the each word of the sentence.

3.2 Feature Extraction

There are mainly three different categories of features that can be extracted from argumentation text. Other features are obtained by combining these three features and studying their performance. Each of them is discussed below.

Word2vec Word2vec was proposed by Mikov [11]. There are two different ways of implementation - a continuous bag of words (CBOW) and Skip gram. In the case of CBOW, word vector is predicted from the context of adjacent word whereas skip-grams is the inverse of CBOW in which context words is predicted from the given words. We used Wikipedia dump of 05-02-2016 as an input to the Word2vec implementation of Gensim, where we generate 100 dimension vector for each word. From our training set, we take each word from the sentence and then find the corresponding vector among the generated word vectors. And then we take the average of all vectors of the words present in the sentence and consider it as the sentence vector.

Unigram The bag of words approach is used to represent the document; all numbers are mapped to the same token and TF-IDF to the normalized the unit length is used to weight the words. This function is calculated as

$$tf - idf(w_i, d) = tf(w_i, d) ln \frac{N}{df(w_i)} \tag{6}$$

where $tf(w_i d)$ is the frequency word w_i in document $d, df(w_i)$ is the number of documents where w_i appears and N is the number of documents in the collection.

Sentence Position Sentence position is a reciprocal function of the position of the respective sentence. The respective sentence scores the highest and gradually decrease up to the final sentence that scores lowest.

$$Score_n = \frac{1}{n} \tag{7}$$

These features are manipulated with the text processing technique; Wordnet and Conference-resolution to generates new features.

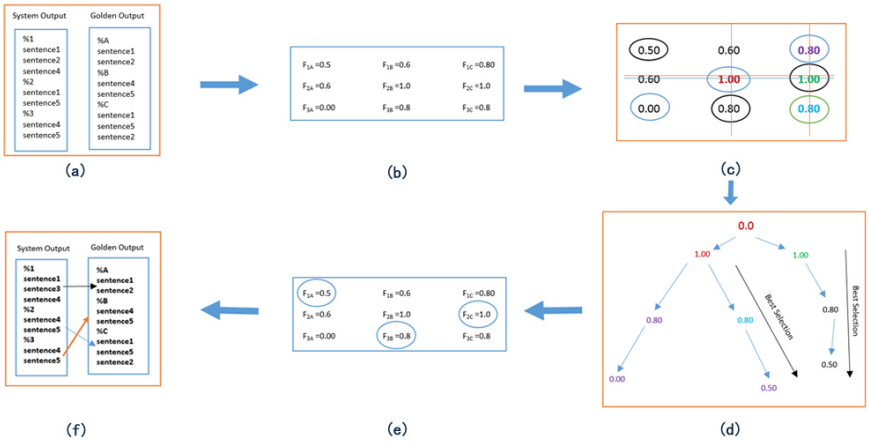


Fig. 1. F-measure score before and after applying ACIA

3.3 Experiments

We used Fuzzy c-mean (FCM) clustering Algorithm proposed by Bezdek [10]. The number of clusters is provided according to the number of arguments present in the corpus. We define fuzziness value $m \in \{1.1, 1.3, 2.0\}$. Distribution of sentence to the respective cluster Algorithm (DSCA) is proposed to transform the membership value generated by FCA in the cluster form. Next, we developed "Appropriate Cluster Identification Algorithm (ACIA)" that helps to select the best mapping between our system's cluster and the golden standard clusters. The idea behind the algorithm is to identify the appropriate index position/cluster respect to the golden standard data sets. The algorithm works accordingly: F-measure value is calculated between the i^{th} cluster of the system and the j^{th} cluster of the golden standard data sets. In the figure 1 we consider 3×3 matrix; in (c) maximum f-measure value is selected to freeze it and the procedure repeat again for other remaining values; in (d) Nodes are connected with the cost value $c = 0$ to form a tree structure. The total cost of each route is calculated. The route that scores maximum value is selected; in (f) the closest cluster/argument is selected as defined in (e).The f-measure value is calculated between the closest cluster/argument as selected; Furthermore, the average f-measure value is calculated as described in section 2.2.

The affect of applying the ACIA algorithm can be seen in the figure 2. The figure shows the comparative f-measure value obtained before and after applying ACIA algorithm. We can observe that the value of "After ACIA "is above 0.3 for all files whereas in the case of "Before ACIA "the maximum value is 0.3. However, we have found some limitations in the algorithm. For instance, time complexity is very high: $O(n \times n)$.

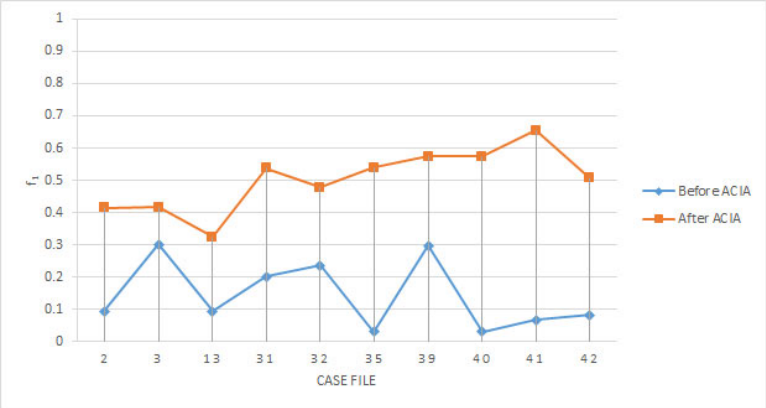


Fig. 2. F-measure score before and after applying ACIA

4 Results and Evaluation

The experiment was conducted with all the features that are listed in section 3.2 with the parameters of fuzziness value of FCM algorithm $m \in \{1.1, 1.3, 2.0\}$ and threshold value $t \in \{0.1, 0.001, 0.0001, 0.00001\}$. The result that we obtained from the combined features of Word2vec, TFIDF and Sentence Position in threshold value (t) = 0.00001 and FCM fuzziness (m) = 1.3. Since the number of arguments is unknown for the Fuzzy Clustering algorithm, we take 11 Case Laws (files) from the golden standard data set that has arguments from 4 to 8. We ran the experiments by assigning 4 to 10 arguments for each file. The file that has 4 arguments only scored 0.655 f-measure value which is the highest value. In contrast, the file that has 9 arguments scored the lowest f-measure value 0.255. From this result, we can consider that as the number of argument increases the performance of the system decreases. However, the approach that we proposed can be used for any kinds of corpora and is not limited to any specific domain.

5 Conclusion and Future Work

We presented a clustering technique proposal to group arguments in legal cases. Overall, the results that we achieved are at a satisfactory level. The average f-measure of system prediction in the files that have 4 to 8 arguments is 0.510.

As a future work, we will add more features such as semantic similarities to improve the results. We also plan to reduce the time complexity of the ACIA algorithm, allowing us to analyze corpora that have a higher number of arguments. Moreover, as an extension of this work, we plan to tag these argumentative sentences in each cluster/argument either as a premise or conclusion.

Acknowledgment

The current work is funded by EMMA-WEST 2013 in the framework of the EU Erasmus Mundus Action 2.

References

1. Douglass R Cutting, David R Karger, Jan O Pedersen, and John W Tukey. Scatter/gather: A cluster-based approach to browsing large document collections. In *Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 318–329. ACM, 1992.
2. Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H Witten. The weka data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1):10–18, 2009.
3. Anna Huang. Similarity measures for text document clustering. In *Proceedings of the sixth new zealand computer science research student conference (NZCSRSC2008), Christchurch, New Zealand*, pages 49–56, 2008.
4. John Lawrence, Chris Reed, Colin Allen, Simon McAlister, Andrew Ravenscroft, and David Bourget. Mining arguments from 19th century philosophical texts using topic based modelling. In *Proceedings of the First Workshop on Argumentation Mining*, pages 79–87. Citeseer, 2014.
5. George A Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995.
6. Raquel Mochales and Marie-Francine Moens. Argumentation mining. *Artificial Intelligence and Law*, 19(1):1–22, 2011.
7. Raquel Mochales-Palau and M Moens. Study on sentence relations in the automatic detection of argumentation in legal cases. *FRONTIERS IN ARTIFICIAL INTELLIGENCE AND APPLICATIONS*, 165:89, 2007.
8. Christian Stab and Iryna Gurevych. Annotating argument components and relations in persuasive essays. In *Proceedings of the 25th International Conference on Computational Linguistics (COLING 2014)*, pages 1501–1510. Dublin City University and Association for Computational Linguistics, August 2014.
9. Prakash Poudyal, Teresa Goncalves and Paulo Quaresma. Experiments On Identification of Argumentative Sentences In *Proceeding of 9th International Conference on Software, Knowledge, Information Management and Applications (SKIMA) Chengdu, China*, 2016.
10. James C. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms*. Kluwer Academic Publishers, Norwell, MA, USA, 1981.
11. Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.
12. Ricardo Baeza-Yates and Berthier Ribeiro-Neto. *Modern information retrieval*, volume 463. ACM press New York, 1999.
13. M.E.S. Rodrigues and L Sacks. A scalable hierarchical fuzzy clustering algorithm for text mining. In *Proceedings of the 5th international conference on recent advances in soft computing*, pages 269–274, 2004.

Experiments for Target Oriented Sentiment Analysis in Social Media

Enkhzol Dovdon and José Saias

DI - ECT - Universidade de Évora
Rua Romão Ramalho, 59
7000-671 Évora, Portugal
d36506@alunos.uevora.pt,jsaias@uevora.pt

Abstract. The paper describes a topic-based message polarity classification system. We used several features with sentiment lexicons and NLP techniques, and Maximum Entropy as the classifier algorithm using tweet datasets.

Keywords: target oriented sentiment analysis, opinion mining, sentiment analysis

1 Introduction

Text data has been growing dramatically. There are huge amount of social media on the web, review, forum discussions, blogs and social networks. Thus, we have demands to process and mine from Social networks and online platforms. Sentiment analysis in user-generated content, are valuable for market and trend analysis. Sentiment analysis, also called opinion mining, is the field of study that analyzes opinions of people, sentiments, appraisals, attitudes, and emotions towards entities and their attributes expressed in written text [1]. Processing of sentiment analysis helps us to automatically distinguish from these written opinions.

We work on Target-oriented (aspect-based, target-based, entity based, or topic based) sentiment analysis which focuses on the precise features (aspects) of the sentiment target. This paper describes our system evaluation in target-based message polarity classification that is to classify positive or negative sentiment of a given message (tweet) towards that target (topic). For example: *Given message: "I'm going to the Nike employee store tomorrow and I'm so excited!", Target: "nike", Classified polarity: "positive".*

We utilized a supervised machine learning classifier, having bag-of-word (BoW), lemmas, bigrams of adjective, punctuation based features and a lexicon-based features. The rest of the paper is structured as follows: In Section 2, we present some related work in features and approaches with a lexicon. In Section 3, this section describes the algorithm and feature representation used to detect sentiment of text. In Section 4, the experimental results are introduced. Finally, the conclusions, as well as further work are described in Section 5.

2 Related Work

There are many works associated with the target-oriented sentiment analysis. Some of these works have focused on probability distribution model of particular features and approach. The system of *Sentiue* [2] used a separate MaxEnt classifier of MALLET (MAchine Learning for LanguagE Toolkit) [3] with bag-of-word-like features (lemmas, bigram, presences, etc.) for Aspect based Sentiment Analysis in SemEval-2015 Task 12. Kamps [4] developed a simple distance measure, that focuses almost exclusively on taxonomic relations and WordNet and determined usage of the semantic orientation of adjectives. Pak [5] utilized presence of an n-grams (n=1,2,3), as a binary feature of a BoW representation using TreeTagger. Fong [6] focused on news articles, which tend to use a more neutral vocabulary using MALLET to implement and training six classifiers for sentiment analysis and compared them. Their experimental results show that the Naive Bayes classifier performs the best of six algorithms. Singh [7] achieved an accuracy as 81.14 in their experiments, had implemented two Machine Learning based classifiers (Naive Bayes and SVM), the Unsupervised Semantic Orientation approach with POS tagging and the SentiWordNet approaches for sentiment classification of a huge amount of movie reviews. Their used two combined scheme (Adjective + Adverb) of SentiWordNet [8] approach was very noteworthy.

3 Method

This section describes feature extraction and a classifier of the sentiment analysis for our system. We used the tool MALLET that supports a variety of supervised classifiers, which makes it ideal for a comparative study of our experiences. We developed current system using several valuable ideas from previous work [2] for Target and Aspect based Sentiment Analysis.

3.1 Datasets

In the experiments, we performed for the system were carried out by training and testing our models on datasets generated in editions of previous years of the tasks at SemEval ("International Workshop on Semantic Evaluation") [9] in Table 1 . All tweets are annotated for polarity as a positive or negative by the organizers of SemEval. We have balanced the ratio of per class is 70:30 for training and test datasets.

3.2 Lexicons

SentiWordNet 3.0 We used to extract features with *SentiWordNet 3.0* lexical resource that supports sentiment classification and opinion mining applications. This lexicon based publicly available resource that provides positive

Dataset	All	Positive	Negative
Training set	13271	10464	2807
Test set	5687	4484	1203
Total	18958	14948	4010

Table 1. Training and test sets of the system.

and negative scores for words. An example of synset terms: *"word:easy#1, positive score:0.625, negative score:0.25, text:an easy job; an easy problem; an easy victory"* Another example of synset terms: *"word:easy#12, positive score:0.25, negative score:0.625, text: obtained with little effort or sacrifice, often obtained illegally; easy money"*

PolarizedWordsList 1.0 Some words appear more than once in *SentiWordNet* lexicon. For an example: "easy", this word is used in 12 different sentences on *SentiWordNet*. In other words, there are 12 use cases of the word and diverse polarity scores (positive or negative score). Thus, we created *PolarizedWordsList 1.0* lexicon which based on average polarity score which was created using all usage cases of a word in SentiWordNet records. *PolarizedWordsList 1.0* consists of 147306 records such as *"easy(word)< tab >0.28333333(positive score)< tab >0.175(negative score)" and "easy_money(word)< tab >0.0(positive score)< tab >0.125(negative score)"*. In section 4, Experiment result is presented using *PolarizedWordsList 1.0*.

PolarizedWordsList 1.1 The result of the experiment using *PolarizedWordsList 1.0* was unsatisfactory due to the fact that there are incorrect polarity scores where were estimated with average polarity scores. Thus, we have improved our lexicon. 6789 of 147306 words in *PolarizedWordsList 1.0* were replaced with **the lists of positive and negative words** [10] and [11]. Then, we also removed many words that polarity scores of positive and negative were 0. The improved *PolarizedWordsList 1.1* lexicon includes 38812 words such as *"easy(word)< tab >1(positive score)" and "easy_money(word)< tab >-1(negative score)"*. In section 4, Experiment result is shown using *PolarizedWordsList 1.1*.

3.3 Feature extraction

We have performed standard data preprocessing steps on the system of tweets prior to classification. Text preprocessing consists of tokenization, removing all capitalization, stop word removal, POS tagging and lemmatization with Stanford CoreNLP [12] and MALLET. An instance was created for each tweet text which includes extracted features. Some features are used additional the lexicon resources such as PolarizedWordsList 1.0 and 1.1. We utilized several combinations of features. Polarized term was based on PolarizedWordsList 1.1 in Features combination 1 to 2.

Features combination-1. The below features to extract from each instance were:

- polarized term for lemmas of nouns, verbs, adjectives, and adverbs before and after target position.

For example:
Target: *"aaron rodgers"*; Tweet: *"today is the day we have been waiting for all summer: the day we see aaron rodgers on the field again. so excited."*; Extracted features:
(1) `#PREV.VBZ.be.positive` for "is",
(2) `#PREV.NN.day.positive` for "day",
(3) `#PREV.VBG.wait.negative` for "wait",
(4) `#PREV.VB.see.positive` for "see",
(5) `#NEXT.NN.field.positive` for "field",
(6) `#NEXT.JJ.excited.positive` for "excited".

If any of the previously mentioned lemmas appeared before a target position in a tweet, it is chosen as the polarized term feature and set a tag as positive or negative. The words were extracted with lemmatization of Stanford CoreNLP (Nouns: NN, NNS, NNP, NNPS; Verbs: VB, VBD, VBG, VBN, VBP, VBZ; Adjectives: JJ, JJR, JJS; Adverbs: RB, RBR, RBS).

Features combination-1.A The below features to extract from each instance were:

- polarized term for lemmas of nouns, verbs, adjectives and adverbs before target position.

Features combination-1.B The below features to extract from each instance were:

- polarized term for lemmas of nouns, verbs, adjectives and adverbs after target position

Features combination-2 The below features to extract from each instance were:

- polarized term for lemmas of nouns, verbs, adjectives, and adverbs before and after target position,
- BoW with a feature for each lemma before and after target position,
- polarized term for bigram words before and after target position,
- presence of negation terms,
- presence of exclamation/question mark based on adjectives.

Features combination-2.A The below features to extract from each instance were:

- polarized term for lemmas of nouns, verbs, adjectives and adverbs before target position,
- BoW with a feature for each lemma before target position,

– polarized term for bigram words before target position,

– presence of negation terms,

– presence of exclamation/question mark based on adjectives.

Features combination-2.B The below features to extract from each instance were:

– polarized term for lemmas of nouns, verbs, adjectives and adverbs after target position,

– BoW with a feature for each lemma after target position,

– polarized term for bigram words after target position,

– presence of negation terms,

– presence of exclamation/question mark based on adjectives.

Features combination-3 is mostly same to 2. However, there were used four lexicons: PolarizedWordsList 1.0 (1), 1.1 (2), the lists of positive and negative words (3), and **Sentiment Composition Lexicon for Opposing Polarity Phrases (SCL-OPP (4))**[13] for polarized term of the below features combination. If a word is in all lexicons, the system will choose from priority order of 4-3-2-1 lexicons. We also used trigram words for the polarized term and if a word was included in polarized term for trigram words, the word would not use in an individual polarized term for unigram word. For example of some extracted features:
(1) `#bigram.bsclopp.positive` is for "junk food",
(2) `#trigram.asclopp.positive` is for "so freaking cute",
(3) `#bigram.bsclopp.minus` is for "perfect storm",
(4) `#BEFORE.NN.TWO.neutral` is for "fact",
(5) `#BEFORE.NN.ONE.negative` is for "consternation",
(6) `#polExclMark.after.positive` is for "!",
(7) `#AFTER.JJ.TWO.positive` is for "entire".

After this step, each text document in the system will be represented by a feature vector using MALLET.

3.4 Classifier training

The classifier algorithm was Maximum Entropy and the classifier model features were previously mentioned features. MaxEnt seeks the probability distribution model that best fits the features observed in the text. We have trained a classifier with instance list where each tweet text had been created as an instance with feature vector. We used a binary classification (positive or negative) for the training in the system. A single sentence in a tweet may have several sentiment polarities about different aspects. Thus, we tried to consider it in feature selection phase that has to choose correct sensitive words as a feature depends on a target.

4 Results

The classification results using each features combination are presented in Table 2. The results of features focused after target position are higher than the results of features focused before target position. The results of Features combination 1 and 3 are better than others. There is not much difference between F1-scores of positive and negative classes in Features combination-2.

Features combination	Class label	Precision	Recall	F1
1	positive	0.846	0.494	0.624
	negative	0.261	0.665	0.375
	average	0.553	0.579	0.499
1.A	positive	0.795	0.694	0.741
	negative	0.226	0.333	0.270
	average	0.510	0.513	0.505
1.B	positive	0.835	0.659	0.737
	negative	0.289	0.516	0.370
	average	0.562	0.587	0.553
2	positive	0.828	0.237	0.369
	negative	0.223	0.816	0.350
	average	0.525	0.526	0.359
2.A	positive	0.795	0.465	0.587
	negative	0.217	0.552	0.311
	average	0.506	0.508	0449
2.B	positive	0.840	0.361	0.505
	negative	0.238	0.743	0.360
	average	0.539	0.552	0.432
3	positive	0.850	0.483	0.616
	negative	0.261	0.682	0.378
	average	0.555	0.582	0.497

Table 2. Results of experiments

5 Conclusions

We have presented an approach that incorporates the MaxEnt with various features to solve the target-based message polarity. Our system is part of first author's work on text classification, included in PhD ongoing work.

Our system was performed 0.486 F1 score, and other systems were achieved from 0.285 to 0.89 F1 scores in SemEval-2017 [14]. The evaluation became a good experience for us. In this paper, the results of our experiments were from 0.359 to 0.553 F1 scores using several lexicons and feature combinations.

We think there are some reasons to decrease a result of current system such as unbalanced datasets (positive tweets are greater than negative tweets), all

text converted to lower case (upper case text is also important) and Polarized-WordsList 1.1 consists of many wrong scores of polarity.

Many people usually use an entirely different language on social media sites such as Twitter and Facebook. Thus, we will focus on social media and informal language learning. As further work we propose the following:

- push state-of-the-art,
- improve the current features,
- use more features,
- use more lexicons such as AFINN [15] and NRC Emoticon [16],
- explore different techniques that can be used in target-oriented sentiment analysis.

Acknowledgments. This work was supported by EACEA under the Erasmus Mundus Action 2, Strand 1 project gLINK - Sustainable Green Economies through Learning, Innovation, Networking and Knowledge Exchange. We would also like to thank the LabInterop project, for providing the infrastructure. LabInterop is funded by *Programa Operacional Regional do Alentejo* (INALENTEJO).

References

1. Liu, B. (2015). Sentiment analysis: Mining opinions, sentiments, and emotions. Cambridge University Pre.
2. Saias, J. (2015, June). Sentiue: Target and aspect based sentiment analysis in semeval-2015 task 12. Association for Computational Linguistics.
3. McCallum, A. K. (2002). Mallet: A machine learning for language toolkit.
4. Kamps, J., Marx, M., Mokken, R. J., & De Rijke, M. (2004, May). Using WordNet to Measure Semantic Orientations of Adjectives. In LREC (Vol. 4, pp. 1115-1118).
5. Pak, A., & Paroubek, P. (2010, May). Twitter as a Corpus for Sentiment Analysis and Opinion Mining. In LREc (Vol. 10, No. 2010).
6. Fong, S., Zhuang, Y., Li, J., & Khoury, R. (2013, August). Sentiment analysis of online news using MALLET. In Computational and Business Intelligence (ISCBI), 2013 International Symposium on (pp. 301-304). IEEE.
7. Singh, V. K., Piryani, R., Uddin, A., & Waila, P. (2013, January). Sentiment analysis of textual reviews; Evaluating machine learning, unsupervised and SentiWordNet approaches. In Knowledge and Smart Technology (KST), 2013 5th International Conference on (pp. 122-127). IEEE.
8. Baccianella, S., Esuli, A., & Sebastiani, F. (2010, May). SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining. In LREC (Vol. 10, pp. 2200-2204).
9. Nakov, P., Ritter, A., Rosenthal, S., Sebastiani, F., & Stoyanov, V. (2016). SemEval-2016 task 4: Sentiment analysis in Twitter. Proceedings of SemEval, 1-18.
10. Hu, M., & Liu, B. (2004, August). Mining and summarizing customer reviews. In Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 168-177). ACM.

11. Liu, B., Hu, M., & Cheng, J. (2005, May). Opinion observer: analyzing and comparing opinions on the web. In Proceedings of the 14th international conference on World Wide Web (pp. 342-351). ACM.
12. Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J. R., Bethard, S., & McClosky, D. (2014, June). The stanford corenlp natural language processing toolkit. In ACL (System Demonstrations) (pp. 55-60).
13. Kiritchenko, S., & Mohammad, S. M. (2016). Sentiment composition of words with opposing polarities. In Proceedings of NAACL-HLT (pp. 1102-1108).
14. Rosenthal, S., Farra, N., & Nakov, P. (2017). SemEval-2017 Task 4: Sentiment analysis in Twitter. Proceedings of the 11th International Workshop on Semantic Evaluation.
15. Nielsen, F. . (2011). A new ANEW: Evaluation of a word list for sentiment analysis in microblogs. arXiv preprint arXiv:1103.2903.
16. Mohammad, S. M., & Turney, P. D. (2010, June). Emotions evoked by common words and phrases: Using Mechanical Turk to create an emotion lexicon. In Proceedings of the NAACL HLT 2010 workshop on computational approaches to analysis and generation of emotion in text (pp. 26-34). Association for Computational Linguistics.

Análise de Sentimentos: Revisão do Estado da Arte

Luís Rosário

Universidade de Évora, Portugal
d5696@alunos.uevora.pt

Resumo. A Análise de Sentimentos ou *Opinion Mining* é o estudo computacional dos sentimentos, opiniões, avaliações, atitudes e emoções de pessoas em relação a certas entidades (indivíduos, questões, eventos, tópicos ou seus atributos, etc.). Com o surgimento de grandes quantidades de textos de opinião *online*, oriundas das mais diversas fontes, tornou-se possível o acesso direto e rápido a um grande conjunto de opiniões, com o intuito de identificar o seu conteúdo e classificar os sentimentos aí expressos.
Este documento tem como objetivo apresentar as diversas abordagens e o trabalho realizado para Análise de Sentimentos até à data, com questões relativas a este campo e perspetivas de trabalhos futuros.

Palavras-chave. Análise de Sentimentos, *Opinion Mining*, Análise de Subjetividade

1 Introdução

No mundo de hoje, as informações textuais disponíveis podem ser classificadas basicamente em duas grandes categorias: factos e opiniões. Os factos representam as afirmações objetivas sobre as entidades. Opiniões são as afirmações subjetivas que refletem os sentimentos e a perceção de uma pessoa sobre uma determinada entidade.

As opiniões têm uma grande influência sobre o comportamento das pessoas, empresas, instituições e governos. Estas são importantes e ajudam no processo de tomada de decisão. Por exemplo, numa perspetiva meramente comercial, a opinião é muito relevante para o consumidor, quando este pensa adquirir um bem ou serviço. Por outro lado, as empresas têm todo o interesse estratégico em tomar conhecimento da opinião dos seus clientes ou potenciais clientes.

A *World Wide Web* trouxe consigo uma grande quantidade e variedade de textos de opinião, oriundas de fontes como: redes sociais, *blogs*, fóruns, jornais *online*, *sites* para avaliação de produtos e serviços, etc. Esta vasta informação constitui uma oprtunidade e, ao mesmo tempo, um repto para a Análise de Sentimentos. Uma oportunidade para se afirmar como uma solução mais adequada, em termos de rapidez e custos, relativamente aos métodos tradicionais (inquéritos e sondagens). E um repto, porque é necessário definir novos métodos e construir novas ferramentas para processar automaticamente o conteúdo das publicações.

Para realizar essas tarefas, investigadores e académicos estão há uma década e meia a trabalhar rigorosamente na área de Análise de Sentimentos, também chamada de *Opinion Mining* ou de Análise de Subjetividade. Não é um problema trivial!

O presente artigo tem como objetivo apresentar uma visão geral do estado da arte da Análise de Sentimentos. Após esta introdução, o ponto 2 irá descrever as etapas e respetivas abordagens desta área. O ponto 3 refere os principais desafios deste campo. O ponto 4 remete para as conclusões e para o trabalho futuro.

2 Etapas da Análise de Sentimentos

As etapas principais da Análise de Sentimentos são: a Identificação, a Classificação de Sentimentos e a Sumarização [1]. Para além destas, há etapas preliminares como a Aquisição de Dados e o Pré-processamento.

2.1 Aquisição de Dados

Atualmente há uma grande variedade e quantidade de recursos *online* que podem ser utilizados na Análise de Sentimentos. A aquisição de dados está, por isso, muito dependente de cada fonte e do respetivo formato. Algumas empresas, como é o caso do *Twitter* e do *Facebook*, disponibilizam APIs (*Application Programming Interface*) para acesso a dados. São exemplos: o Twitter REST API que disponibiliza dados estáticos como o perfil do utilizador, o *Twitter Streaming API* e o *Twitter4J API* que disponibilizam *streaming* de *tweets* e o *Facebook Graph API* que disponibiliza informação sobre nós, campos e relações. O recurso a motores de pesquisa também é uma opção válida.

2.2 Pré-processamento

Antes de se avançar para a Análise de Sentimentos propriamente dita, os dados brutos obtidos podem precisar de ser pré-processados. Algumas das técnicas mais usadas nesta etapa são: criação de *tokens*, remoção de *stop words*, marcação POS (*Part-Of-Speech*) e *Stemming*. A criação de *tokens* é utilizada para decompor o texto em cada termo que o compõe. Os delimitadores utilizados para esta tarefa geralmente são: o espaço em branco entre os termos, quebras de linhas, tabulações e alguns caracteres especiais. A *stop words* é uma lista de termos não representativos de um texto e por esse motivo são removidos. A marcação POS é uma marcação gramatical, permitindo adicionar conhecimento sobre os termos do texto. *Stemming* é o método para redução de um termo ao seu radical. Com a sua utilização, os termos derivados de um mesmo radical serão contabilizados como um único termo.

2.3 Identificação

Esta etapa consiste em encontrar, num conjunto de textos obtidos, as entidades, os seus possíveis aspetos e associá-los ao respetivo conteúdo subjetivo (sentimentos). A

identificação das entidades, aspetos e sentimentos depende da granularidade escolhida para análise. O nível de granularidade está dependente do contexto e da área de aplicação, podendo surgir ao nível do documento, da palavra, do aspeto, da frase, entre outros, como descrito em [2].

A complexidade da identificação das entidades resulta também da fonte escolhida e do seu grau de estruturação. A área de aplicação mais usual na Análise de Sentimentos é a Avaliação de Produtos e Serviços, porque a entidade pode ser mais facilmente identificada [3]. Já no que diz respeito a notícias, *blogs* ou *posts* não se conhecem as entidades abordadas, podendo envolver várias na mesma porção de texto.

Para além do já referido, esta etapa pode incluir a extração e a seleção de aspetos. A extração de aspetos é uma tarefa difícil, pois os dados textuais podem apresentar muito ruído e este apresentar-se de forma dispersa. A seleção de aspetos também é crucial para o sucesso da Análise de Sentimentos e utiliza vários tipos de técnicas diferentes, nomeadamente *Pointwise Mutual Information* (PMI), Qui-quadrado e Indexação Semântica Latente, entre outras [2].

No que diz respeito à identificação de sentimentos, estes podem ser regulares ou comparativos; diretos ou indiretos; implícitos ou explícitos. A maioria dos estudos concentram-se em sentimentos regulares, diretos e explícitos, por serem mais fáceis de ser tratados.

2.4 Classificação de Sentimentos

A Classificação de Sentimentos, ou de Polaridade é frequentemente um problema binário, ou seja, classifica um determinado texto em duas classes: positivo ou negativo. Esta classificação pode ser ainda mais pormenorizada, contendo mais graus de intensidade, por exemplo: muitoPositivo, moderamentePositivo. Outra situação, é considerar a classe neutra que engloba textos sem uma polaridade clara.

Abordagens para a Classificação de Sentimentos.
As abordagens de classificação de Sentimentos podem ser divididas em três grandes categorias: a) Abordagens baseadas em Aprendizagem Automática; b) Abordagens baseadas em Léxico; c) Abordagens Híbridas. A qualidade do modelo preditivo, destas abordagens, é medida em termos de *Accuracy*, Precisão, *Recall* ou Medida-F1 [4].

Abordagens baseadas em Aprendizagem Automática.
O objetivo principal destas técnicas é descobrir automaticamente regras gerais em grandes conjuntos de dados, que permitam extrair informações implícitas. Estas técnicas podem ser divididas em dois tipos: Aprendizagem Supervisionada e Aprendizagem não Supervisionada. A grande distinção entre elas é o facto de que na Aprendizagem Supervisionada é adquirido um modelo de classificação com base num corpus de treino previamente rotulado, o que não acontece na Aprendizagem não Supervisionada.

A Aprendizagem Supervisionada, por sua vez, pode ser feita com recurso a: Árvores de Decisão, Classificação baseada em Regras, Classificação Linear (que inclui *Support*

Vector Machines e Redes Neuronais) e Classificação Probabilística (que inclui Máxima Entropia, *Bayesian Network* e *Naive Bayes*) [5].

Abordagens baseadas em Léxico.
Estas abordagens dependem de léxicos de sentimentos (compilações de palavras ou expressões de sentimentos associadas à respetiva polaridade) e podem ser de dois tipos: com base em Dicionário ou com base em Corpus. A Abordagem baseada em Dicionário começa com palavras de sentimentos iniciais e, em seguida, pesquisa o dicionário, à procura dos seus antónimos ou sinónimos. Por outro lado, a Abordagem baseada em Corpus começa com uma lista de sentimentos iniciais e, em seguida, encontra outro sentimento num grande corpus com o objetivo de descobrir palavras de sentimento no contexto. Esta abordagem pode utilizar um método estatístico ou semântico para determinar a polaridade do sentimento.

Abordagens Híbridas.
Como o próprio nome indica, estas abordagens resultam da utilização conjunta das abordagens anteriormente referidas. Acrescente-se ainda que nesta situação a vertente léxical é dominante e habitualmente é acompanhada por uma técnica de Aprendizagem Supervisionada.

2.5 Sumarização

O objetivo desta etapa é poder identificar o sentimento médio ou que prevalece num grupo de pessoas sobre uma determinada entidade. Para a grande quantidade de sentimentos é necessário criar métricas e sumários que permitam quantificar e agregar a diversidade de sentimentos encontrados, a respeito de uma mesma entidade. São assim criadas métricas que representam um sentimento geral.

3 Desafios

O problema da Análise de Sentimentos é sobretudo um problema de análise textual. A seguir são apresentados alguns desafios nesta área.

3.1 Problema da Negação

A negação desempenha um papel importante na alteração da polaridade do adjetivo associado e, portanto, a polaridade do texto. Uma solução possível para lidar com palavras de negação (como *não*, *nem* e *nunca*) é inverter a polaridade do adjetivo que surge após uma palavra de negação, por exemplo, "*O restaurante é bom*" (deve ser classificado como positivo), "*O restaurante não é bom*" (deve ser classificado como negativo). No entanto, esta solução não serve para casos como, "*Não admira que o restaurante seja bom*" e "*Não só a comida era saborosa, como também o serviço e o espaço eram excelentes*". O uso de técnicas de processamento de linguagem pura ou o

uso puro de modelos matemáticos não conseguem abordar completamente as negações [6].

3.2 Problemas de Contexto

Certas palavras exibem polaridades diferentes quando usadas em domínios distintos. Por exemplo, "*O filme foi inspirado num livro duvidoso*" tem orientação negativa, no entanto, "*Eu inspirei-me no romance*" tem orientação positiva. Aqui, a palavra "*inspirar*" exibe duas polaridades diferentes para dois contextos distintos. A Análise de Sentimentos é geralmente realizada visando um domínio específico e isso tem mostrado bons resultados de precisão. Mas um analisador de sentimentos generalizado ainda permanece um desafio devido aos diferentes sentidos de uma palavra ou frase em domínios distintos [6].

3.3 Resolução de Pronomes

Certas palavras que possam conter sentimentos e que sejam referenciadas por um pronome como *ele*, *isto*, *isso*, *ele*, *ela*, etc. exigem tarefas complexas para determinar o objeto referenciado, introduzido anteriormente.

3.4 Generalização da Linguagem

É necessário um dicionário distinto para cada idioma diferente e para cada domínio diferente. Até momento, os analisadores de sentimentos foram implementados apenas para a língua inglesa. Um analisador de linguagem geral seria benéfico, pois daria uma visão ampla do sentimento em relação a uma entidade.

3.5 Conhecimento do Mundo

Os textos podem conter outras entidades para se referir à entidade em análise. Neste caso, o conhecimento da entidade que é usada para se referir a outra, é necessário para a correta identificação do sentimento. Por exemplo, "*Ele é tão rápido como o Usain Bolt*". Aqui, para se determinar a orientação sentimental do texto, é preciso saber quem é *Usain Bolt*.

3.6 Mapeamento de Gírias

As gírias são geralmente formas curtas e informais de palavras originais. Por exemplo, *lol* é uma gíria usada para a expressão "*rir à gargalhada*". Estas palavras não fazem parte de dicionários tradicionais, mas são encontradas frequentemente em textos *online*. Se as gírias pudessem ser mapeadas para palavras originais, os resultados da Análise de Sentimentos poderiam ser melhorados.

3.7 Problemas associados à Ironia e ao Sarcasmo

O uso de ironia ou sarcasmo, onde o sentimento explícito é exatamente oposto ao sentimento expresso explicitamente, constitui mais um obstáculo à Análise de Sentimentos.

4 Conclusões

Dada a grande quantidade de dados textuais não estruturados existentes atualmente, a Análise de Sentimentos tornou-se uma área de crescente interesse. A mesma poderá contribuir para o desenvolvimento de melhores produtos, serviços e melhor qualidade de gestão empresarial.

Este documento tentou mostrar uma visão ampla sobre o trabalho feito até à data, na área da Análise de Sentimentos.

Com base na análise dos artigos lidos, conclui-se que o aperfeiçoamento dos algoritmos de Classificação de Sentimentos ainda é um campo aberto de investigação.

As técnicas mais usadas na Classificação de Sentimentos por ordem decrescente são: *Support Vector Machines*, abordagens baseadas em dicionários, *Naive Bayes*, Redes Neuronais, Árvores de Decisão, Máxima Entropia e outros [2].

O interesse em outras línguas, para além do inglês e da língua chinesa, no campo da Análise de Sentimentos é também notória, pois ainda há falta de recursos disponíveis.

As fontes como *microblogs*, *blogs*, fóruns e fontes de notícias apresentam uma enorme quantidade de informações sobre os sentimentos e opiniões das pessoas, que tem sido utilizadas pela Análise de Sentimentos. Apesar disto ainda é necessária uma investigação mais profunda.

Foram já descritos os principais desafios desta área, que revelam a necessidade de continuar a aperfeiçoar as ferramentas de Processamento de Linguagem Natural. Embora, certos algoritmos tenham obtido bons resultados, ainda não existe uma técnica completa, até agora, que possa resolver todos os desafios.

Referências

1. K. Becker e D. Tumitan, "Introdução à Mineração de Opiniões: Conceitos, Aplicações e Desafios," em Lectures of the 28th Brazilian Symposium on Databases, 2013.
2. K. Ravi e V. Ravi, "A survey on opinion mining and sentiment analysis: tasks, approaches and applications," Knowledge Based Systems, 2015.
3. R. Piryani, D. Madhavi e V. Singh, "Analytical mapping of opinion mining and sentiment analysis research during 2000-2015," Information Processing & Management, vol. 53, nº 1, pp. 122-150, 2017.
4. K. Ahmed, N. E. Tazi e A. H. Hossny, "Sentiment Analysis over Social Networks: An Overview," em 2015 IEEE International Conference on Systems, Man, and Cybernetics, 2015.
5. W. Medhat, A. Hassan e H. Korashy, "Sentiment analysis algorithms and applications: A survey," Ain Shams Engineering Journal, vol. 5, pp. 1093-1113, dec 2014.

6. A. Kaur e N. Duhan, “A survey on sentiment analysis and opinion mining,” International Journal of Innovations & Advancement in Computer Science, vol. 4, pp. 107-116, 2015.