RESOLUÇÃO DE ENTIDADES UTILIZANDO MULTIDÕES

Jacson Hwang

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em Engenharia de Sistemas e Computação, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia de Sistemas e Computação.

Orientador(es): Jano Moreira de Souza

Rio de Janeiro

Outubro de 2014

RESOLUÇÃO DE ENTIDADES UTILIZANDO MULTIDÕES

Jacson Hwang

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA (COPPE) DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM ENGENHARIA DE SISTEMAS E COMPUTAÇÃO.

Examinada por:

_____

Prof. Jano Moreira de Souza, Ph.D.

_____

Prof. Geraldo Bonorino Xexéo, D.Sc.

_____

Prof.ª Ana Maria de Carvalho Moura, D.Sc.

RIO DE JANEIRO, RJ - BRASIL
OUTUBRO DE 2014

Hwang, Jacson

Resolução de Entidades Utilizando Multidões/ Jacson Hwang. – Rio de Janeiro: UFRJ/COPPE, 2014.

XIV, 121 p.: il.; 29,7 cm.

Orientador: Jano Moreira de Souza

Dissertação (mestrado) – UFRJ/ COPPE/ Programa de Engenharia de Sistemas e Computação, 2014.

Referências Bibliográficas: p. 100-106.

1. Crowdsourcing. 2. Resolução de Entidade. I. Souza, Jano Moreira de. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia de Sistemas e Computação. III. Título.

Agradecimentos

Agradeço primeiramente a Deus, Senhor e salvador da minha vida.

Agradeço também Hwang Shun Li e Chen Ching Yueh, aqueles que me geraram e tem cuidado de mim até os dias de hoje.

Agradeço aos meus irmãos André Hwang e Ivan Hwang que me auxiliam a todo instante.

Ao Professor Jano, orientador excepcional, pela oportunidade e paciência na realização deste trabalho, que espero fazer jus a confiança dedicada a mim.

A Sérgio Rodrigues e Marcio Antelio, meus coorientadores, pela disposição e boa vontade em me ajudar em todas as horas, cada um ajudando na forma como pode.

Aos professores Geraldo Bonorino Xexéo e Ana Maria de Carvalho Moura, por aceitarem fazer parte da banca examinadora, abrindo mão de seus compromissos e sacrificando o tempo de suas agendas.

Aos companheiros de trabalho que auxiliaram na construção do sistema.

A todos aqueles que contribuíram diretamente e indiretamente com a pesquisa realizada, parte fundamental deste trabalho.

Resumo da Dissertação apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

RESOLUÇÃO DE ENTIDADES UTILIZANDO MULTIDÕES

Jacson Hwang

Outubro/2014

Orientador: Jano Moreira de Souza

Programa: Engenharia de Sistemas e Computação

É notável a facilidade da criação de novas bases de dados, tornando esta, que era uma tarefa difícil, algo trivial. É comum uma instituição criar múltiplas bases de dados que mantém referências às mesmas entidades no mundo real, como pessoas, produtos e endereços, por exemplo. Ou seja, é evidente a redundância de informações em diferentes fontes de dados. Isto pode acarretar inconsistências indesejadas e é um empecilho em atividades posteriores de mineração de dados e *Business Intelligence*. Resolução de Entidade é um processo que determina se duas referências ao mundo real são, na realidade, referências à mesma entidade. *Crowdsourcing* é o ato de reunir um grupo de pessoas desconhecidas a fim de realizar diversos tipos de tarefas. Essas tarefas podem exigir inteligência, raciocínio lógico, expertise e criatividade, tornando a multidão uma forte candidata na resolução de entidades. Desta forma, o objetivo deste trabalho é propor um modelo capaz de explorar e potencializar a capacidade da multidão sem comprometer a qualidade dos dados, fator essencial para a resolução de entidades.

Abstract of Dissertation presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

ENTITY RESOLUTION USING CROWD

Jacson Hwang

October/2014

Advisor: Jano Moreira de Souza

Department: Systems and Computer Engineering

It is remarkable the simplicity of the creation of new databases, turning this task, which was a difficult one, into something trivial. It is common for an institution to create multiple databases that maintain references to the same entities in the real world, such as people, products and addresses, for example. That is, it is evident the redundancy of information in different sources of data. It can cause undesirable inconsistencies and it is a hindrance to later activities of data mining and Business Intelligence. Entity Resolution is a process that determines if two references of the real world are, actually, references to the same entity. Crowdsourcing is the act of gathering a group of unknown people in order to realize many types of tasks. These tasks may require intelligence, logical reasoning, expertise and creativity, turning the crowd into a strong candidate in the entity resolution. This way, the objective of this work is to propose a model capable of exploring and potentiating the capability of the crowd without compromising the quality of the data, essential fact to the entity resolution.

Sumário

Lista de Figuras

Lista de Tabelas

Lista de Abreviaturas

CSV – Comma-Separated Values

EM – Entity Matching

ER – Entity Resolution

ERA – Entity Resolution Activity

ERM – Entity-Relation Model

ETL – Extract Transform Load

HIT – Human Intelligence Tasks

HOC – Hands-off Crowdsourcing

IRE – Identity Resolution Engine

LOD – Linked Open Data

MDM – Master Data Management

MVC – Model-view-controller

OO – Orientado a Objetos

OYSTER – Open sYSTem Entity Resolution

RL – Record Linkage

SERF –Stanford Entity Resolution Framework

SBGD – Sistema de Gerenciamento de Banco de Dados

SBGDR – Sistema de Gerenciamento de Banco de Dados Relacional

SOG – Synthetic Occupancy Generator

SR – Revisão Sistemática

TWI – Talburt-Wang Index

UTI – Unstructured Textual Information

XML – eXtensible Markup Language

1.Introdução

O grande crescimento no volume de dados na Web, bem como a popularização de atividades extremamente dependentes de dados, tais como: integração dos dados, mineração dos dados, *business intelligence*, fazem com que se dê cada vez mais importância à qualidade desses dados.

Atualmente, a informação é cada vez mais vista como um ativo organizacional (REDMAN, 2008) que não só conduz os processos operacionais, mas de onde pode ser extraído conhecimento (CHAN *et al.*, 2009) usado para melhorar o desempenho organizacional e ajudar a organização a ganhar vantagem competitiva no mercado.

Dentre os problemas que afetam a qualidade dos dados, a duplicidade de registros ocorre de maneira frequente. Resolução de Entidades tem como objetivo primário identificar referências que indicam o mesmo objeto no mundo real. A forma mais comum para executar o processo de busca às referências duplicadas ocorre através de algoritmos computacionais otimizados para tal tarefa.

Os algoritmos estão longe da perfeição quando o assunto é tratar referências que representam o mesmo objeto do mundo real. É nesse momento que a sabedoria da multidão pode auxiliar nessas inúmeras possibilidades de comparação (SUROWIECKI, 2006, LEVY, 2007). Adotando o conceito de computação humana (VON AHN, 2005), apenas o poder de processamento humano é capaz de resolver determinadas tarefas.

Os trabalhos que procuram relacionar a multidão com a resolução de entidades focam principalmente na abordagem híbrida, envolvendo técnicas e algoritmos com o esforço da multidão. Tal abordagem foca em escalabilidade, uma vez que a maior parte dos dados é processada computacionalmente.

Entretanto, é possível observar que nas abordagens propostas a multidão tem sua capacidade bastante limitada, de modo que somente são criadas tarefas de complexidade extremamente baixa. Este trabalho propõe uma abordagem em que a multidão se torne mais presente no processo de resolução de entidades. Esta atitude possibilita tornar o modelo mais genérico de forma a abranger mais situações, como detectar referências equivalentes a partir de referências que não necessariamente têm exatamente os mesmos atributos. Tal liberdade ainda permite a multidão avaliar referências que contenham outros tipos de mídias, como áudio e vídeo, por exemplo. Entretanto, esta liberdade em um primeiro momento, contribui para a diminuição da

qualidade dos dados, porém o modelo proposto foi projetado para atentar a essa questão tão importante.

1.1.Objetivos

Os objetivos deste trabalho são:

1. Criar um modelo que permita maior participação da Multidão na Resolução de Entidades possibilitando a entrada de dados de diversas bases e com diferentes atributos agregando estratégias de qualidades.
2. Avaliar a eficiência do modelo utilizando métricas de qualidade adequadas ao contexto de resolução de entidades.
3. Elaborar uma forma de aplicação do modelo de fácil entendimento e uso por administradores de dados.

1.2.Limitações do estudo

O trabalho, embora realize uma abordagem de resolução de entidades, é limitado, pois não contempla todas as atividades de resolução de entidades descritas na seção 3.4. O modelo de processo sugerido tem seu foco na etapa principal de ER (ERA3), que consiste na busca e agrupamento de referências que indiquem a mesma entidade chamadas de referências equivalentes.

Outra limitação é a escalabilidade do processo deste trabalho. Como a abordagem do processo envolve particularmente a multidão, isto é, não é envolvido nenhum algoritmo computacional de filtragem de referências para diminuir a quantidade de registros. As velocidades de resoluções dependem da quantidade de usuários que estão realizando as tarefas. Entretanto, essa abordagem tem como objetivo principal a eficácia e não a velocidade somente.

1.3.Organização do trabalho

O Capítulo 1 apresenta o escopo do problema a ser tratado, assim como as suas limitações. A organização do trabalho e os objetivos a serem alcançados pela pesquisa também são retratados nesse capítulo.

O Capítulo 2 apresenta uma revisão de literatura sobre um dos principais temas dessa dissertação: *crowdsourcing*. Outro tema indiretamente relacionado a área, porém essencial para este estudo, a Qualidade de Dados associado a Multidão, serão revisados neste capítulo.

O Capítulo 3 aborda o tema central deste trabalho, apresentando conceitos fundamentais de resolução de entidades. Além disso, apresenta as técnicas e algoritmos mais utilizados nessa área. Ao fim deste capítulo é abordado alguns trabalhos que aplicaram sistemas *crowdsourcing* no contexto de resolução de entidades.

O Capítulo 4 descreve o modelo de processo que engloba a resolução de entidades com estratégias de qualidade de dados proposto por esse trabalho. O capítulo segue descrevendo a maneira como o modelo pode ser aplicado em casos reais. Além disso, a ferramenta CERM é descrito como prova funcional do modelo.

O Capítulo 5 contempla o relato dos experimentos realizados, incluindo a justificativa para a utilização das métricas de desempenho, base de dados e estratégias de qualidade de dados no modelo.

O Capítulo 6 traz a conclusão do trabalho com relato dos objetivos alcançados e trabalhos futuros que esta dissertação poderá gerar.

Na parte final deste trabalho, são apresentados as Referências Bibliográficas e os Apêndices da pesquisa.

2.Sabedoria das Multidões

Em seu livro, *The Wisdom of Crowds*, SUROWIECKI (2005) utiliza o termo homólogo pela primeira vez e argumenta que um grupo se comporta de maneira impressionantemente inteligente, se colocado sob circunstâncias apropriadas. Ainda mais, afirma que estes grupos, na maioria das vezes, são mais inteligentes do que um indivíduo sozinho no seu melhor estado (SUROWIECKI, 2006). O autor, de maneira geral, define a sabedoria das multidões como uma inteligência coletiva, que é construída a partir da agregação de avaliações individuais sob condições adequadas.

Segundo LÉVY (1998), inteligência coletiva é como "[...] uma inteligência distribuída por toda parte, incessantemente valorizada, coordenada em tempo real, que resulta uma mobilização efetiva das competências". O autor ainda afirma que o principal objetivo dessa inteligência é o reconhecimento e o enriquecimento mútuo (LÉVY, 1998).

Este capítulo tem como objetivo explorar e expor as principais características relacionadas à sabedoria da multidão e o conceito de *Crowdsourcing*. Também será abordado como a qualidade da multidão influencia no resultado final de uma tarefa, e como o *design* deve ser construído a fim de otimizar o esforço da multidão. O capítulo é encerrado com as considerações gerais de como este trabalho adota algumas das estratégias apresentadas.

2.1.Características das multidões

Algumas condições devem ser atendidas para garantir a sabedoria da multidão: diversidade, independência e descentralização. A seguir, as três características serão detalhadas:

2.1.1.Diversidade

O conhecimento consolidado por um grupo heterogêneo é mais significativo porque cada membro pode contribuir com diferentes perspectivas sobre um mesmo assunto, enquanto em grupos homogêneos cada membro contribui muito pouco para o todo pelo fato de serem muito próximos (MARCH, 1991).

PAGE (2007, p.162) afirma que, em condições específicas, "[...] um grupo de solucionadores aleatórios de problemas supera em desempenho um grupo constituído pelos melhores solucionadores"[1] . Essas condições específicas reforçam a ideia de que nem todo grupo aleatório conseguiria executar algo como projetar uma plataforma de petróleo, por exemplo.

O conceito de diversidade pode ser divido ainda em: diversidade de identidade, habilidades e postura política (BRABHAM, 2007). A identidade, por exemplo, seria o gênero, raça, nacionalidade ou religião.

As diversidades nas habilidades específicas contribuem para a resolução de problemas complexos. À medida que um grupo recebe membros com as mais variadas habilidades, maiores são as chances de o grupo resolver problemas de diferentes níveis e áreas de conhecimento (BRABHAM, 2007).

A última diversidade é bem específica, pois diz respeito à solução de problemas referentes a grupos políticos diferentes.

2.1.2.Independência

Decisões coletivas têm melhores resultados quando tomadas por pessoas diferentes e de formas independentes, fundamentadas principalmente em opiniões pessoais (SUROWIECKI, 2006 p.86). A independência contribui para a consolidação das informações de modo que as pessoas não sejam influenciadas por outras e, além disso, contribui para que erros cometidos não se propaguem.

Ainda mais, quanto mais as pessoas se utilizam de sua própria experiência e opinião, maiores são as chances de agregar conhecimentos diversificados para o grupo.

---

[1] Tradução do autor para: "[...] a randomly selected collection of problem solvers outperforms a collection of the best individual problem solvers."

2.1.3.Descentralização

SUROWIECKI (2006) afirma que a melhor solução para pessoas orgulhosas, egoístas e tímidas trabalharem de forma ótima é através de um sistema descentralizado.

O motivo é devido a esses sistemas criarem um ambiente favorável em que pessoas com dificuldades de trabalho em equipe cooperem sem restrições.

Um sistema descentralizado tem como ponto forte a opinião dos indivíduos, diferentemente de um sistema centralizado, onde as ordens e decisões são acatadas por todos e as opiniões pessoais não tem tanta relevância.

Uma desvantagem dessa abordagem é a inexistência de garantias de que uma informação importante encontrada em uma parte do sistema seja disponibilizada para o restante do sistema.

2.2.Crowdsourcing

Jeff Howe e Mark Robinson (HOWE, 2006a) criaram o termo *Crowdsourcing*, entretanto o termo foi utilizado pela primeira vez por HOWE (2006b). O termo é composto por duas palavras, *crowd* (multidão) e *outsourcing* (terceirização), que em linhas gerais pode ser interpretado como a terceirização de tarefas para a multidão solucionar.

Uma definição mais formal, dada por um dos criadores (HOWE, 2006a) do termo seria "[...] o ato de uma companhia ou instituição escolher uma função desenvolvida por um empregado e terceirizá-la para uma rede de pessoas indefinidas (e geralmente grande) na forma de chamado aberto"[2].

Segundo BRABHAM (2008), *crowdsourcing* é um "[...] modelo estratégico para atrair uma multidão interessada e motivada de indivíduos capazes de prover

[2] Tradução do autor para: "[...] the act of a company or institution taking a function once performed by employees and outsourcing it to an undefined (and generally large) network of people in the form of an open call."

soluções superiores em qualidade e quantidade em comparação às soluções tradicionais oferecidas por empresas"[3].

HOWE (2008) afirma ainda que a área de *crowdsourcing* está emergindo como um meio economicamente viável de intensificação de esforços que requerem intervenção humana.

Uma definição mais abrangente é a adotada por QUINN & BEDERSON (2011), que define *crowdsourcing* como o ato de explorar as habilidades de pessoas com o objetivo de atingir um bom resultado como, por exemplo, a resolução de um problema ou apoiar a tomada de uma decisão. Consideramos esta definição mais adequada para ser relacionada aos outros conceitos abordados neste trabalho.

2.2.1.Características

Existe uma enorme dificuldade em classificar os padrões de práticas de *crowdsourcing*. Alguns exemplos são as diferentes formas que cada autor utiliza para realizar essa avaliação (GEIGER *et al.*, 2011, QUINN & BEDERSON, 2009, HOWE, 2009).

Esta seção tem como objetivo classificar as mais diversas variedades de sistemas de *crowdsourcing* disponíveis.

2.2.1.1.Tipos

A classificação dos tipos de *crowdsourcing* varia de autor para autor. SCHENK & GUITTARD (2011) classificam os sistemas em três tipos, de acordo com as tarefas: simples, complexas e criativas.

Em geral, as tarefas simples são coletas de dados, categorização de produtos, não necessitando de especialização por parte do usuário. As tarefas complexas necessitam de conhecimento específico e em geral oferecem recompensas monetárias. As tarefas criativas exigem certo nível de abstração e criatividade para serem realizadas.

---

[3] Tradução do autor para: "[...] strategic model to attract an interested, motivated crowd of individuals capable of providing solutions superior in quality and quantity to those that even traditional forms of business can".

Outra abordagem é a classificação segundo a natureza de contribuição do usuário, utilizado por HOWE (2009). Elas não são mutuamente exclusivas, sendo elas: *crowdfunding*, voto, criatividade e inteligência coletiva.

No trabalho proposto, iremos adotar o conceito de tarefa simples para, justamente, permitir que o maior número de usuários possa participar das tarefas geradas.

2.2.1.2.Competências dos indivíduos

A competência do indivíduo é uma importante característica para um sistema de *crowdsourcing*. Os critérios adotados para selecionar os usuários dependem essencialmente do propósito do sistema. Alguns sistemas necessitam de candidatos com conhecimentos específicos, outros, não.

QUINN & BEDERSON (2011) definem quatro tipos de pré-seleção: (1) sistemas que exigem habilidade ou conhecimento mínimo, (2) sistemas construídos para um nicho bem específico, por exemplo, para uso interno da empresa, (3) ambas as abordagens anteriores e (4) nenhuma restrição.

GEIGER *et al.* (2011b) afirmam que a diversidade de usuários pode comprometer a qualidade dos dados; é por isso que a avaliação da competência do usuário é essencial para determinados sistemas.

2.2.1.3.Visibilidade de contribuições

GEIGER *et al.* (2011) classificam em quatro tipos a visibilidade que os usuários têm através do sistema em relação às contribuições de outros indivíduos:

- Nenhuma: o usuário não tem qualquer permissão para visualizar outras contribuições;
- Visualizar: um usuário pode apenas visualizar a contribuição de alguém;
- Avaliar: o usuário pode avaliar a contribuição de outra pessoa;
- Modificar: o usuário pode alterar a contribuição de outro.

É importante notar que a falta de visibilidade ou excesso dela pode comprometer a qualidade dos dados. Alguns sistemas, como o reCAPTCHA (Seção 2.2.3.4), não utilizam nenhum nível de visibilidade, porém, caso utilizasse pelo menos o segundo

nível mais restrito (Visualizar), a estratégia de forçar o usuário a transcrever a imagem seria inútil. O Wikipedia (WIKIPEDIA, 2014), por exemplo, permite a modificação da contribuição de outros usuários.

2.2.1.4.Motivação

A motivação é fator fundamental para manter a multidão em constante realização das tarefas, além de influenciar na qualidade de dados. A seguir serão apresentadas as maneiras como os sistemas motivam os seus usuários:

Recompensa

A monetização é a maneira mais comum adotada pelos grandes sistemas de *crowdsourcing* como fator motivacional. Inclusive, diversas pessoas utilizam esses sistemas como forma de complementação de renda (CHEN & DOLAN, 2011). Entretanto, casos de trapaças também são mais comuns, uma vez que os grandes sistemas mantêm o anonimato dos trabalhadores (QUINN & BEDERSON, 2011).

Dois exemplos de grandes sistemas seriam o Amazon Mechanical Turk[4] e o CrowdFlower[5], que monetizam cada tarefa executada pelos trabalhadores. Em geral os preços giram em torno dos centavos.

A recompensa é mais ampla do que somente a monetização. Exemplos como o *Gift cards*, muito comum nos Estados Unidos, ou dinheiro virtual em jogos também são frequentes neste cenário (QUINN & BEDERSON, 2011).

GEIGER *et al.* (2011) classificam a monetização em dois tipos: (1) fixa, quando o valor da remuneração é uniforme, e (2) dependente, quando as contribuições são avaliadas, e as de maior impacto são mais valorizadas.

Diversão

É muito comum as pessoas passarem seu tempo em diversos tipos de entretenimentos na internet. A variedade é enorme e vai desde leitura de blogs, visualização de vídeos até jogos online (QUINN & BEDERSON, 2011).

---

[4] https://www.mturk.com/mturk/welcome

[5] http://www.crowdflower.com

Aproveitando este fato, VON AHN (2005) criou diversos jogos que exploram o passatempo das pessoas para produzir dados úteis.

Outra área que explora esse nicho são os Jogos Com Propósito (*Games With a Purpose*), onde os jogos são na realidade uma *interface* para a contribuição de sistemas de *crowdsourcing* já existentes. Os principais motivos para as pessoas jogarem é justamente a diversão que os jogos proporcionam. O ESP Game (Seção 2.2.3.2) e o CAPTCHINO (Seção 2.2.3.5) são Jogos Com Propósito que deram certo.

Altruísmo

Problemas importantes e que têm impacto na sociedade e no convívio como um todo atraem a atenção de pessoas altruístas. É comum as pessoas estarem contribuindo em sistemas de *crowdsourcing* apenas pelo simples fato de quererem ajudar. A recompensa nesses casos é a satisfação em contribuir para a resolução de um problema de objetivo maior (QUINN & BEDERSON, 2009).

Reputação

QUINN & BEDERSON (2011) afirmam que grandes organizações podem motivar as pessoas a participarem de seus sistemas apenas por reputação, sem ao menos serem recompensados financeiramente. Além da monetização através das propagandas, os usuários enviam vídeos ao Youtube (YOUTUBE, 2014) também em busca de fama e reputação, o que acaba gerando milhares de novos vídeos todos os dias (YUEN *et al.*, 2011).

2.2.2.Qualidade no contexto de multidão

Até alguns anos atrás, a preocupação principal era a resolução do problema de maneira rápida e barata, e isso a multidão tinha e tem condições de oferecer. Entretanto, a rapidez e o custo reduzido não são as únicas questões que influenciam na adoção da multidão. A qualidade dos dados que são produzidos é tão, ou se não, mais importante do que os outros fatores.

Muitas experiências publicadas indicam que a qualidade não acompanha a escalabilidade (WAIS *et al.*, 2010). Ou seja, quanto maior é o volume de dados envolvido, a qualidade de dados produzidos pela multidão é diminuída.

O custo e tempo para se verificar os resultados submetidos por trabalhadores são comparáveis à execução da tarefa em si (IPEIROTIS *et al.*, 2010). Ainda mais, os autores afirmam que caso sejam necessários 10 trabalhadores na execução de uma tarefa, o custo tende a ser comparável à solução de especialistas. A redundância exagerada aumenta significantemente os custos de soluções utilizando multidão.

Há alguns anos, autores estão avaliando o custo benefício na utilização de *crowdsourcing* em ambientes pagos. Um deles foi o trabalho de SNOW *et al.* (2008), que utiliza o AMT (*Amazon Mechanical Turk*) para gerar anotações linguísticas, como reconhecimento afetivo e ordenação temporal de eventos. Para cada tarefa, as respostas dos trabalhadores eram anotadas e sua qualidade mensurada baseadas nas respostas geradas por especialistas.

As plataformas de *crowdsourcing* oferecem oportunidades sem precedentes na criação de benchmarks de avaliação, entretanto sofrem de um mal: a qualidade dos dados que varia de acordo com o trabalhador, e consequentemente do seu nível de competência e de seus desejos (KAZAI *et al.*, 2011).

Os trabalhadores são bem diversificados, podem ser diferentes no quesito sócio econômico ou em habilidades e motivações. Todos esses fatores influenciam o resultado final de uma tarefa, tornando o controle da qualidade uma tarefa bastante difícil (ROSS *et al.*, 2010).

Como resultado da diversidade da multidão, a qualidade dos dados pode ser baixa, além da desonestidade, a aleatoriedade e o descuido intensificarem esta situação. Diversos trabalhos já foram escritos com base na detecção desses tipos de trabalhadores. Alguns analisam o tempo gasto nas tarefas (DOWNS *et al.*, 2010, KITTUR *et al.*, 2008, SNOW *et al.*, 2008) enquanto outros comparam os resultados obtidos com um conjunto já conhecido de respostas (ALONSO & BAEZA-YATES, 2011, KAZAI, 2011).

A seguir, é apresentada uma divisão simples de como a qualidade dos dados em sistemas de *crowdsoucing* é analisada atualmente pela imensa maioria de trabalhos relacionados nessa área.

2.2.2.1.Perfil

KAZAI *et al.* (2011) observou um padrão de comportamento de diversos *turkers* baseado no tempo de execução de um HIT (*Human Intelligence Tasks* – Tarefas de

Inteligência Humana), a acurácia e a quantidade de anotações úteis. A partir dessas observações, foram definidos cinco perfis de trabalhadores:

- *Spammer*: são trabalhadores maliciosos que não produzem qualquer dado útil para a tarefa.
- Descuidado: se preocupam pouco com a qualidade do seu trabalho. Estes trabalhadores podem contribuir com uma elevada fração de respostas úteis, mas trabalham muito rápido, gastando pouco tempo por HIT. Entretanto, a consequência desse comportamento é a baixa precisão.
- Diligente: são caracterizados por serem cautelosos ao realizar a tarefa, alto nível de contribuição, tempo gasto maior que a média e alta acurácia dos dados.
- Incompetente: contribuem bastante na tarefa, e embora gastem um tempo considerável por HIT, produzem apenas com baixa qualidade, muitas das vezes devido à falta de habilidade ou competência, ou até à má compreensão da tarefa.
- Competente: são trabalhadores qualificados que têm alto nível de contribuição e com alta precisão, além de trabalharem rápido, tornando-os trabalhadores muito eficientes e eficazes.

Tabela 1 – Distribuição do perfil de trabalhadores (adaptado de KAZAI *et al.* (2011))

	Spammer	**Descuidado**	**Incompetente**	**Competente**	**Diligente**
Resultados úteis	Baixo	Alto	Alto	Alto	Alto
Tempo médio	-	Baixo	Alto	Baixo	Alto
Acurácia	-	Baixo	Baixo	Alto	Alto

VUURENS *et al.* (2011) dedicam seu trabalho a identificação dos diferentes tipos de perfil de trabalhadores, com foco na categoria dos *spammers*. O primeiro tipo a ser comentado são os trabalhadores éticos[6], aqueles que contribuem para resultados precisos, que seguem as instruções e destinam-se a produzir resultados significativos.

[6] Tradução do autor para: "ethical workers"

Entretanto, LE *et al.* (2010) relatam que até mesmo os trabalhadores éticos podem produzir resultados de baixa qualidade. Este baixo rendimento pode ser justificado pela não compreensão total da intenção do solicitante da tarefa, ou pela incapacidade do trabalhador (VUURENS et al. 2011).

Estes trabalhadores éticos de baixo rendimento são chamados de trabalhadores desleixados[7] e aqueles com melhores resultados de trabalhadores adequados[8.]

ZHU & CARTERETTE (2011) realizaram uma análise comportamental e constataram que uma parcela mostrou um padrão de votação rápida e alternada. Estes trabalhadores mostram uma capacidade comportamental de trapaça ao tentar escolherem respostas aleatórias, de modo que seria difícil ao solicitante descobrir estas desonestidades. Ainda em VUURENS *et al.* (2011), é descrito um grupo que possui precisão média de acertos, indicando assim uma aleatoriedade nas respostas. Este grupo é intitulado como *spammers* aleatórios[9] e se utilizam dessa técnica para não serem detectados.

Outro grupo descrito por ZHU & CARTERETTE (2011) se comporta de forma uniforme. Estes trabalhadores não têm interesse em cumprir a tarefa de forma correta, entretanto, não se utilizam de técnicas avançadas de trapaça, respondendo sempre as mesmas respostas. Estes trabalhadores são chamados de *spammers* uniformes[10] (VUURENS et al. 2011). Embora os padrões se repitam, sendo facilmente detectados por inspeção manual, a detecção automatizada pode não perceber devido a outros *spammers* uniformes responderam o mesmo ao longo de muitas questões.

E finalmente, através do experimento de ZHU & CARTERETTE (2011), notou-se que três dos trabalhadores que apresentavam suspeitas de serem *spammers* aleatórios mostraram uma precisão média de 0,52, o que tornava improvável que os trabalhadores escolhessem todas as questões aleatoriamente. Tal atitude indica que estes trabalhadores responderam algumas questões diligentemente; talvez as questões fáceis ou para etapa de qualificação. Estes trabalhadores são chamados *spammers* semi-aleatórios[11] (VUURENS et al. 2011).

---

[7] Tradução do autor para: "sloppy workers"

[8] Tradução do autor para: "proper workers"

[9] Tradução do autor para: "random spammers"

[10] Tradução do autor para: "uniforme spammers"

[11] Tradução do autor para: "semi-random spammers"

2.2.2.2.Design de Tarefas

A partir da classificação de perfil, KAZAI *et al.* (2011) sugerem que o design das tarefas pode ser desenvolvido pensando nas características de cada tipo de trabalhador.

KITTUR *et al.* (2008) reforça que o *design* de tarefa influencia na qualidade dos dados. Os autores propuseram duas tarefas para avaliar a qualidade de 14 artigos do Wikipedia. No primeiro *design*, os colaboradores estavam livres para avaliar os artigos e para preencher uma caixa de texto com as melhorias necessárias. Neste contexto, metade dos dados foi classificada como suspeita (respostas maliciosas).

Em outro momento, o segundo design de tarefas consistia de quatro questões com resultados já conhecidos antes da avaliação por parte da multidão, e ainda foi desenvolvido de modo que a multidão tivesse recompensas para respostas corretas. Com esta abordagem somente 2,5% dos dados foi classificada como suspeita.

Portanto, o design de tarefa é sim um importante fator que auxilia no aumento da qualidade de dados. Projetar tarefas em que o custo para trapaceá-las é maior do que para completá-las é uma estratégia para desestimular a trapaça.

Até mesmo usuários motivados podem trapacear no sistema, especialmente quando a recompensa é monetária ou envolve competição. A falta de entendimento, como comentado na seção anterior, pode provocar a inserção de dados não qualificados ao propósito. QUINN & BEDERSON (2009, 2011) definem algumas estratégias de controle de qualidade:

- Checagem automática: para alguns problemas, algoritmos são capazes de realizar uma verificação da resposta do usuário, permitindo saber se o trabalhador respondeu ou não corretamente;
- Modelo econômico: é a recompensa monetária de acordo com a qualidade da contribuição;
- Revisão em multicamada: grupos produzem os resultados, enquanto outros grupos da multidão revisam as contribuições;
- Tarefas defensivas: são aquelas em que é muito mais complexo trapacear que realizar a tarefa propriamente dita;
- Redundância: é a disponibilização das tarefas repetidas vezes, obtendo assim, diversas respostas. Em geral, é realizado um sistema de votação, através da

maioria, para eleger a solução ótima. Essa forma de *design* contribui para diferenciar trabalhadores de alta e baixa qualidade.

- Implantação de *gold label*: consiste em inserir uma quantidade pequena de tarefas onde as respostas já são conhecidas, permitindo a identificação de usuários de baixa qualidade ou trapaceiros.
- Filtro estatístico: filtrar ou agregar os dados e descobrir *outliers* para serem removidos;

Design de tarefas que propiciam uma relação duradoura com o trabalhador pode ser interessante, pois é criado uma relação de confiança em ambas as partes. As pessoas são propensas a se especializarem em tarefas com treinamento melhorando seu desempenho ao longo do tempo. Consequentemente isso acarreta em um controle de qualidade menor. Além disso, o retorno da opinião dos trabalhadores melhora cada vez mais o design na tarefa e os próprios trabalhadores podem treinar novatos (CHEN & DOLAN, 2011).

Manter um meio de comunicação efetiva e estabelecer um sistema de recompensa para os trabalhadores também estimulam os trabalhadores a produzirem bons resultados e em grande volume (CHEN & DOLAN, 2011).

Uma forma de projetar o sistema *crowdsourcing* é a criação de etapas de qualificação para filtrar os trabalhadores com menos eficiência. Existem duas maneiras para realizar uma qualificação: a manual e a automatizada. A forma manual, embora seja mais exata, é bem mais demorada e demanda a presença de pessoas para avaliação. A outra maneira é a realização automática do processo, CHEN & DOLAR (2011) recomendam rodadas de qualificação automatizadas processadas por algoritmos, entretanto esta abordagem necessita de gabarito dos resultados previamente cadastrados no sistema. Outra vantagem da qualificação automatizada é a maior escalabilidade em relação à forma manual.

A fim de obter julgamentos mais precisos a respeito dos resultados da multidão, votos para a mesma tarefa são reunidas. Estas tarefas podem ser agregadas em uma única resposta por um algoritmo de consenso, onde o mais comum nesses casos é a votação por maioria (RAYKAR *et al.*, 2010). Embora possa dar bons resultados, esse algoritmo é bem criticado, pois presume que todos os trabalhadores tenham a mesma

qualidade de produção, enquanto a qualidade, como pode ser vista nas seções anteriores, pode ser bastante diversificada.

A exemplo de rodadas de qualificação automatizadas, o experimento realizado por WAIS *et al.* (2010) contemplou 4660 participantes, dos quais apenas 79 atingiram um nível alto de acurácia suficiente para participarem das tarefas reais.

Outra estratégia bastante utilizada no controle de trabalhadores a longo prazo é o sistema multicamada de pagamento (NOVOTNEY & CALLISON-BURCH, 2010, CHEN & DOLAN, 2011). A aplicação desse tipo de sistema auxilia na retenção de bons trabalhadores, resultando em uma força de trabalho altamente qualificada (CHEN & DOLAN, 2011). Essa abordagem ainda permite pagamentos mais elevados para os trabalhadores mais "graduados", evitando o desperdício de dinheiro com trabalhadores ruins.

A técnica de multicamada comentada por (CHEN & DOLAN, 2011) consiste em diferenciar o pagamento de acordo com a qualidade do trabalhador. No trabalho em questão, foram criadas duas etapas com tarefas idênticas. Para a primeira etapa, o trabalhador recebia uma recompensa de $0.01, já para a segunda recebia $0.05. A medida visa motivar o trabalhador para que produza resultados melhores. A troca de nível era realizada de tempos em tempos de forma manual, o que tornava o processo lento e pouco escalável.

2.2.3.Aplicações

2.2.3.1.Amazon Mechanical Turk

AMT (*Amazon Mechanical Turk*) é um sistema barato e rápido que coleta anotações de um ampla base de contribuidores pagos não especializados da Internet (SNOW *et al.*, 2008).

Lançado em 2005, o AMT permite a distribuição de microtarefas para uma multidão de usuários resolvê-las. Os usuários que pagam pelo serviço, também chamados de *requesters*, dividem suas tarefas em formas de HIT, para que os usuários monetizados (*turkers ou workers*) realizem as tarefas (ROSS *et al.*, 2010).

A plataforma abriga diversos tipos de tarefas, que variam desde categorização de imagens à pesquisa de mercado. O tempo e o valor monetário variam de acordo a tarefa: quanto mais complexa, maior é a recompensa financeira (ROSS *et al.*, 2010).

Ao contrário de muitas plataformas, o AMT não concentra tarefas em apenas uma área específica; ele procura oferecer uma variedade de problemas e com isso mais *requesters* podem solicitar os serviços, fazendo com que mais *turkers* possam contribuir, oferecendo um ambiente propício para a prática de *crowdsourcing*.

2.2.3.2.ESP Game

ESP Game é um jogo que funciona com base na disputa entre duas pessoas escolhidas aleatoriamente. Elas jogam ao mesmo tempo e o objetivo é escrever termos que representem uma imagem. Os jogadores não têm nenhuma informação sobre o outro. Após ambos digitarem a mesma palavra, a figura é alterada e o jogo prossegue.

Uma concordância é alcançada no momento em que um termo é escrito por ambos os jogadores, cuja palavra torna-se uma descrição daquela imagem (VON AHN & DABBISH, 2004).

Uma forma para evitar que a mesma descrição seja utilizada diversas vezes é fazer com que após um número N de concordâncias, a palavra seja exibida ao lado da figura e não possa mais ser inserida. Este processo tornou-se extremamente efetivo, dado que algoritmos computacionais não têm a mesma eficiência se comparados a uma pessoa para rotular imagens (VON AHN, 2009).

2.2.3.3.GalaxyZoo

O projeto GalaxyZoo foi concebido devido à necessidade de categorizar cerca de 50.000 imagens de galáxias para a criação de um banco de dados (FORTSON *et al.*, 2011). A partir dessa experiência, surgiu a necessidade de classificar as imagens de forma que levasse menos tempo.

Primeiramente, o voluntário aprende a categorizar as imagens a partir de um tutorial; após esta etapa é permitida a classificação das diversas imagens ainda não categorizadas por astrônomos. A primeira versão do GalaxyZoo consistia na classificação da galáxia em elípticas, colisões e espirais, e caso fosse espiral ainda era necessário informar a direção dos braços (GALAXY ZOO, 2014). A segunda versão foi inspirada na descoberta da capacidade da então recém multidão de voluntários. Esta versão consistia em classificar mais de 200.000 galáxias de acordo com o número de braços espirais, entre outras informações (GALAXY ZOO, 2014). A terceira versão, GalaxyZoo: Hubble, tinha como objetivo compreender a evolução das galáxias, adicionando outras bases de dados maiores em relação a projetos anteriores

(FORTSON *et al.*, 2011). A quarta versão, lançada em 2012, conta com novas câmeras instaladas no telescópio Hubble, e com fotos mais detalhadas e profundas do universo (GALAXY ZOO, 2014).

2.2.3.4.reCAPTCHA

Criado por VON AHN *et al.* (2008), é uma variação do CAPTCHA, que de forma implícita auxilia na transcrição de livros.

O CAPTCHA (acrônimo para *Completely Automated Public Turing test to tell Computers and Humans Apart*) é uma imagem contendo caracteres distorcidos que aparecem na parte inferior em formulários Web (VON AHN *et al.*, 2003). Além disso, ele é utilizado para prevenir que programas autômatos abusem de serviços online.

Enquanto o CAPTCHA exibe imagens com caracteres aleatórios gerados por computador, o reCAPTCHA exibe palavras de textos digitalizados (VON AHN *et al.*, 2008). Para diferenciar a utilização de um humano para uma máquina, o reCAPTCHA exibe duas palavras, onde uma delas é conhecida e a outra não é. Esta última é retirada de algum livro ou figura, sendo que não é possível um computador transcrevê-la. Em resumo, a multidão acaba por ajudar na transcrição de livros de forma explícita.

2.2.3.5.CAPTCHINO

Captchino[12] (SAHA *et al.*, 2012) é uma ferramenta gamificada desenvolvida para fazer uma análise da eficiência dos métodos geração de *captchas* existentes na atualidade. Os usuários passam por uma fase de cadastro onde dizem o seu sexo e idade. O jogo é composto de seis rodadas e em cada uma delas é apresentado um *captcha* que utiliza algum dos métodos atual.

Os resultados foram analisados de acordo com a idade e o sexo das pessoas e a taxa de usabilidade dos *captchas* foi descrita como a porcentagem de acertos dos usuários. Os resultados mostraram que os métodos mais eficazes foram o (1) *Combocaptcha*, onde duas ou mais palavras são exibidas ao usuário e este deve escolher uma ou mais opções que contenham aquelas palavras e (2) *Claptcha* (SAHA *et al.*,

[12]http:// www.lpuprojectcaptcha.com

2012), onde um alfabeto é mostrado ao usuário e este deve dizer qual a localidade onde este alfabeto é utilizado.

2.2.3.6.Senses

Senses (VENHUIZEN *et al.*, 2013) foi um jogo desenvolvido para *Wordrobe*[13] com o intuito de ajudar na identificação do significado de palavras em frases. O jogo é constituído de várias rodadas onde são apresentadas frases ao jogador, e o jogo deve escolher uma das opções que melhor descreve o significado da palavra destacada naquela frase. O jogo possui um sistema de aposta, onde o jogador indica uma porcentagem de certeza quanto à resposta dada. Quanto maior a aposta, maior será a quantidade de pontos ganhos pelo jogador caso acerte. Este sistema foi adicionado para que fosse possível avaliar a dificuldade de uma dada questão (VENHUIZEN *et al.*, 2013). Ao longo do jogo o jogador também adquire pontos e *achievements* que decoram seu perfil no jogo.

A consolidação dos resultados é feita utilizando-se o voto da maioria. Nos testes foram recebidas 5.478 questões, com uma média de três respostas de jogadores por questão. Destas questões, foram analisadas 115 que receberam exatamente seis respostas cada uma.

Nos resultados, o sistema se provou bastante eficiente, com taxas de precisão acima de 80% quando analisados todos os dados e acima de 90% quando analisadas somente as respostas com valor de aposta maior que 80%.

[13]http://www.wordrobe.org

2.3.Considerações Gerais

As características de *Crowdsourcing* apresentadas são importantes para o modelo proposto. A diversidade é alcançada no modelo, pois não há restrições quanto ao nível de conhecimento das pessoas. A independência é conquistada uma vez que cada usuário não sabe o que outro realizou. A descentralização, por sua vez, é obtida através da liberdade da resolução de entidade, com o mínimo de restrição possível para a execução da tarefa por parte do usuário.

As tarefas desenvolvidas nesse trabalho são simples com a finalidade de resolver um problema complexo de resolução de entidades. A qualidade é controlada por meio de uma rodada de qualificação que classifica os usuários de acordo com o seu desempenho. Além disso, a avaliação do perfil do usuário é realizada ao longo das rodadas e sua evolução durante as tarefas resulta em uma melhor avaliação de seu perfil, enquanto *spammers* deverão ser penalizados ao decorrer da rodada.

Além disso, alguns elementos como a pontuação e o ranking foram aplicados como fatores motivacionais, gerando assim uma disputa por reputação e diversão. Além disso, conta com o altruísmo das pessoas, pois estarão contribuindo para um trabalho de dissertação de um colega.

3.Resolução de Entidades

Resolução de Entidades[14] (*Entity Resolution* – ER) é uma área de importância crescente para empresas e governos. Muitos pesquisadores têm trabalhado nessa área desde os anos 1950. (ZHOU & TALBURT, 2011).

Para TALBURT (2010, p.29), resolução de entidade "[…] é o processo de determinar se duas referências a objetos do mundo real estão se referindo ao mesmo objeto ou a objetos diferentes."[15]. O autor complementa que o termo entidade descreve tanto um objeto real como uma pessoa, um lugar, ou coisa.

Em TANSEL *et al.* (2006), a entidade pode ser descrita como algo físico, como uma pessoa, ou casa, ou pode ser alguma construção lógica, como uma família, uma rede social ou uma lista de pessoas que gostam de um tipo de gênero musical.

Embora o processo de ER seja aplicado a pares de referências, ER pode ser aplicado a grandes conjuntos de referências, bastando apenas agregar todas as referências de um mesmo objeto em subconjuntos (TALBURT, 2010).

Dentro deste contexto de conjunto de referências, BENJELLOUN *et al.* (2009) definem ER como "o processo de identificação e fusão de registros definidos para representar a mesma entidade no mundo real "[16].

Entidades são descritas em função de atributos. Os valores desses atributos informam sobre as características específicas da entidade. Atributos de Identidade[17] são aqueles em que quando juntos identificam uma entidade da outra. Os atributos como CPF e data de aniversário são atributos de identidade para pessoas, assim como UPC (*Universal Product Code*) seria para produtos.

TALBURT (2010) apresenta o termo Suposição de Referência Única[18], onde afirma que a referência é sempre criada para se referir exclusivamente a uma, e apenas uma entidade.

---

[14] Tradução do autor para: "Entity Resolution"

[15]Tradução do autor para: "[...] is the process of determining whether two references to real-world objects are referring to the same object or to different objects."

[16]Tradução do autor para: "[...] the process of identifying and merging records judged to represent the same real-world entity."

[17] Tradução do autor para: "Identity attributes"

[18] Tradução do autor para: "Unique Reference Assumption"

A razão para esta suposição é que em situações do mundo real, uma referência pode parecer ambígua, ou seja, ela pode se referir a mais de uma entidade ou possivelmente a nenhuma. Por exemplo, um vendedor pode escrever uma descrição do produto em um pedido de vendas, mas caso a descrição esteja incompleta, a pessoa que processar a ordem de compra pode ficar em dúvida sobre qual produto deve ser encomendado. Embora tenha acontecido este problema, a intenção do vendedor era referenciar um produto apenas. (TALBURT, 2010).

Nesse caso, a completude da anotação pode estar comprometida, ou os dados podem estar desatualizados. Os graus das dimensões da qualidade de dados afetam as operações dos processos de ER e produzem resultados melhores ou piores. Esta é uma razão pela qual a ER é tão próxima do campo de Qualidade da Informação.

3.1.Entidade, referência e instância

A área de resolução de entidades trata a questão de instâncias e referências de forma diferenciada. No contexto de ER, as instâncias de uma entidade de MER (Modelo Entidade Relacionamento) não são propriamente a entidade (Princípio 1 de ER proposto por Talburt) e sim uma referência. Isso ocorre porque o contexto de ER contempla que instâncias diferentes possam representar uma mesma entidade no mundo real. Em teoria, essa duplicidade não deveria ocorrer, mas no mundo real e em diversos banco de dados, ocorre até com muita frequência.

Uma instância de um tipo de entidade, como a entidade Autor na Figura 1, é justamente um registro na tabela no banco de dados e que faz referência a algum autor no mundo real. No contexto de ER, nada impede que exista outra instância que referencie o mesmo autor do primeiro caso.

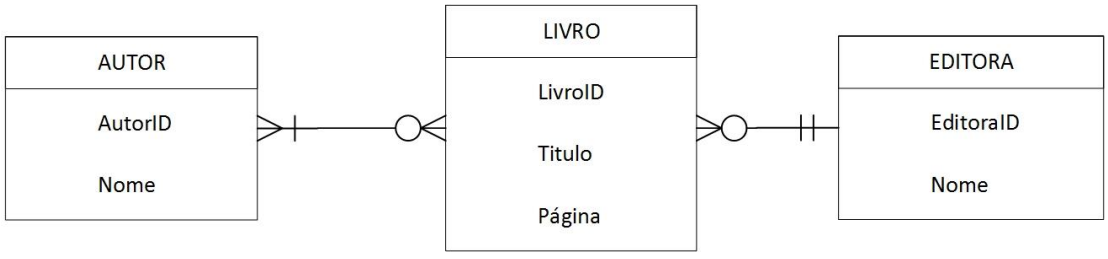


Figura 1 – Exemplo de um simples MER (elaborado pelo autor)

TALBURT (2010) ilustra um exemplo de aplicação de ER em contexto de negócio. O autor supôs uma entidade Cliente e a mesma pode ser referenciada diversas vezes no sistema de informação da empresa, em alguns casos, em sistemas diferentes. Existem muitas razões para que a empresa crie múltiplas referências para um mesmo cliente. Uma delas é que o cliente tenha realizado compras em diferentes canais de vendas ou departamentos. Cada uma tem sua própria base de dados e estas provavelmente não estão integradas com os restantes da empresa.

Outra característica para a proliferação das referências dos clientes no negócio é que as características do cliente, especialmente os seus contatos, mudam a todo momento. Caso o nome, e-mail ou telefone não sejam atualizados corretamente, os sistemas podem assumir que as transações utilizando dados não armazenados representem clientes novos. O reconhecimento desses registros que indicam o mesmo cliente é a essência do ER (TALBURT, 2010).

Em outros casos, o problema pode ser simplesmente a falta de um controle adequado da qualidade de informação em dados de entradas de forma manual, permitindo erros ou variações nos valores. ARKADY (2007) descreve a existência de inúmeros meios os quais podem ser introduzidos erros de qualidade de dados em um sistema de informação.

O reconhecimento de que a informação sobre tipos de entidades críticos no negócio deve ser sincronizada em toda a empresa, deu origem a prática de *Master Data Management* (MDM) (LOSHIN, 2008).

3.2. Terminologia

Uma referência é um conjunto de valores de atributos para uma entidade específica. Quando duas referências indicam a mesma entidade, eles são ditas *co-refer* (CHEN *et al.*, 2009), *matching references* (BENJELLOUN *et al.*, 2009) ou referências equivalentes[19] (TALBURT, 2010).

Ainda em ZHOU & TALBURT (2011), o termo utilizado também é referência equivalente para representar referências que estão relacionadas com o mesmo objeto.

A noção de resolução de entidade surgiu no contexto de remoção de referências equivalentes entre duas listas. FELLEGI & SUNTER (1969) lidavam com este

[19] Tradução do autor para: "*Equivalent reference*"

problema frequentemente e descrevem o processo como *record linking* ou *record linkage*.

Posteriormente, no contexto de SGBD relacional, o foco passou a ser o problema de encontrar e fundir múltiplas instâncias de um mesmo tipo de entidade (HERNÁNDEZ & STOLFO, 1995), tal processo é chamado *merge-purge*. Foi descrito como ER em 2004 em artigos e apresentações de pesquisadores de Stanford, liderado por GARCIA-MOLINA (2006).

TALBURT (2010) argumenta que *record linking* é justamente uma representação de uma decisão de resolução, enquanto ER é um termo mais adequado para descrever todo o processo de decisão ao invés do termo *record linking*. Embora os precursores tenham sido os processos de *merge-purge* e *record linking*, a área de ER tem crescido tanto na parte teórica quanto na parte prática e atualmente descreve uma abordagem muito mais ampla de atividades.

3.2.1.Deduplicação de Dados

O método de Deduplicação de Dados ou *Data Deduplication* consiste no conjunto de mecanismos de compressão de dados que tem por finalidade eliminar a redundância de dados em um determinado conjunto de informações ou bytes.

Com o objetivo de reduzir o espaço utilizado na memória como um todo, este método é amplamente utilizado na identificação de sequências de bytes iguais em um determinado arquivo a ser processado ou transferido para outro local de armazenamento.

Durante o processo de análise do arquivo, trechos de bytes ficam armazenados na memória até o fim da execução do algoritmo, para que trechos de bytes idênticos sejam sumariamente excluídos da sequência de bytes do arquivo.

Existem diferentes tipos de Deduplicação de Dados, como o *Post-process Deduplication* e o *In-line Deduplication*. No primeiro tipo de processo todos os dados são armazenados para que depois se faça uma análise de seu conteúdo para busca de trechos duplicados, fazendo com que a execução e armazenamento do arquivo sejam feitas de forma mais ágil, já que o sistema só analisa seu conteúdo posteriormente. Já no segundo tipo, os dados são armazenados na memória à medida que o sistema analisa suas informações a procura de duplicatas, o que otimiza a quantidade de dados a serem

armazenados na memória, considerando que dados redundantes são eliminados antes mesmo de serem persistidos na memória de um determinado dispositivo.

3.2.2.Record Linkage

O termo foi introduzido amplamente na comunidade científica em 1959 (NEWCOMBE *et al.*, 1959). MALIN & SWEENEY(2005) definem o termo como o processo de busca de entradas relacionadas em uma ou mais relações em uma base de dados para criar uma ligação entre elas[20].

O processo de *Record Linkage*, ou *Data Linkage*, ou ainda *Record Linking*, é simplesmente identificar os grupos de registros equivalentes sem fundi-los. Isto é feito através da atribuição de cada referência no mesmo grupo com um identificador comum chamado de *link*, com grupos diferentes que têm diferentes *link values*. *Linking* é um método para representar as decisões de resolução sobre as referências de entidade, onde, por exemplo, são dadas a duas instâncias de referência (registros) o mesmo *link value* representando a decisão de que eles são referências equivalentes (TALBURT, 2010).

Tal medida evita que haja informações duplicadas em determinado conjunto de dados, economizando espaço de armazenamento e processamento, bem como reduz o tempo gasto na análise das informações existentes.

Para isso, é preciso criar um novo conjunto de dados, removendo duplicatas de informações em um ou mais arquivos ou combinando os mesmos para que o relacionamento de dois ou mais elementos possa ser devidamente analisado e estudado (WINKLER, 2006).

3.2.3.Merge - Purge

O processo de *merge-purge* representa a forma mais básica de ER e inicia com a coleta de todas as referências a serem resolvidas em um único conjunto de dados. Sistematicamente, este processo compara os pares de referências, e aquelas consideradas equivalentes são reunidas em grupos ou *clusters* (TALBURT, 2010).

---

[20] Tradução do autor para: "[...] is the process of finding related entries in one or more related relations in a database and creating links among them."

Geralmente, mantêm-se o melhor exemplar de registro do grupo ou combinam-se os valores dos atributos de todos os registros do grupo para criar um único registro, daí o termo *merge-purge*.

Como nomenclatura análoga para o processo de *Data Linkage*, a técnica de *merge-purge* também caracteriza um processo de correspondência e mistura de dados a fim de reduzir ao máximo o tamanho de um arquivo ou de uma tabela através da exclusão e reaproveitamento de dados idênticos.

Esta nomenclatura é mais comumente atribuída de forma comercial, por aplicações que tem como função e objetivo comprimir arquivos por meio da exclusão de trechos de dados duplicados em um determinado conjunto de dados.

3.3.Princípios de Resolução de Entidades

TALBURT (2010), estabelece aos 7 princípios para o contexto de resolução de entidades:

- Princípio 1 – Sistemas de informação armazenam e manipulam referências às entidades, e não as entidades[21].
- Princípio 2 – ER consiste fundamentalmente em ligar referências equivalentes, e não na correspondência de registros[22].
- Princípio 3 – Falsos negativos em ER são geralmente um problema mais difícil de ser detectado e resolvido do que falsos positivos[23].
- Princípio 4 – Processos de ER são geralmente projetados para evitar falsos positivos ao custo da criação de falsos negativos[24].

[21] Tradução do autor para: "[...] Information systems store and manipulate references to entities, not the entities."

[22] Tradução do autor para: "[...] ER is fundamentally linking equivalent references, not record matching."

[23] Tradução do autor para: "[...] ER false negatives are generally a more difficult problem to detect and solve than are false positives."

[24] Tradução do autor para: "[...] ER processes are generally designed to avoid false positives at the expense of creating false negatives."

- Princípio 5 – Sistemas ER ligam referências através de equivalência deduzida e afirmada. Equivalência deduzida pode ser estabelecida através de combinação direta, equivalência transitiva, ou análise de associação[25].
- Princípio 6 – Resolução de Entidades não é o mesmo que resolução de identidade. Este último é apenas uma forma de ER[26].
- Princípio 7 – Sistemas ER que provêm valores de ligação persistentes devem também implementar alguma forma de gerenciamento de identidade[27].

3.4. Atividades de Resolução de Entidades

TALBURT (2010), estabelece as 5 atividades para a resolução de entidades:

- ERA 1 – Extração de Referência de Entidade. Localizar e coletar referências de entidades de informações não estruturadas.
- ERA 2 – Preparação de Referência de Entidade. Aplicação de perfis, padronização, limpeza de dados e outras técnicas de qualidade de dados para estruturar as referências de entidades anteriores ao início do processo de resolução.
- ERA 3 – Resolução de Referência de Entidade. Determinar se duas referências estão relacionadas à mesma entidade ou entidades diferentes.
- ERA 4 – Gerenciamento de Identidade de Entidade. Construir e manter registros da informação da identidade de entidade ao longo do tempo.
- ERA 5 – Análise de Relação de Entidade. Explorar a rede de associações entre as diferentes, porém relacionadas, entidades.

---

[25] Tradução do autor para: "[...] ER systems link references through inferred and asserted equivalence. Inferred equivalence can be established through direct matching, transitive equivalence, or association analysis."

[26] Tradução do autor para: "[...] Entity resolution is not the same as identity resolution. Identity resolution is only one form of ER."

[27] Tradução do autor para: "[...] ER systems that provide persistent link values must also implement some form of identity management."

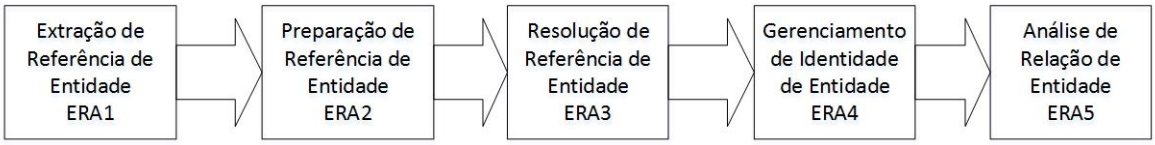


Figura 2 – Em ordem, as Cinco Maiores Atividades de ER (adaptado de TALBURT (2010))

TALBURT (2010) compara o termo ER com o utilizado na área de Tecnologia de Informação (TI), onde este tem dois aspectos: o "Big IT", que significa qualquer coisa com relação a computadores e o sentido mais restrito e técnico, o "Little IT".

Tomadas em conjunto, as cinco atividades apresentadas na Figura 2 incluem o "Big ER." Muitos autores utilizam o termo ER no sentido "Little ER" que compreende apenas o ERA3 ou o ERA2 seguido por ERA3. Por exemplo, o processo de *merge-purge*, que representa a extensão do ER para muitas organizações, é essencialmente uma atividade ERA3. Nem todo processo de ER envolve todas as cinco atividades e diferentes ferramentas e sistemas de ER são projetados para lidar com diferentes atividades no processo global de ER.

3.4.1.ERA 1 – Extração de Referência de Entidade

Esta etapa é essencial quando os dados sobre as referências de entidade são apresentados de forma desestruturada (TALBURT, 2010). Informação estruturada é quando os dados são organizados de uma maneira em que todos os valores de atributos são descritos de forma consistentes e padronizados.

No caso de formatos não estruturados, é necessária uma forma de padronização dos dados para que se possa seguir com a execução das etapas futuras. Quando os dados não estão estruturados, torna muito difícil a identificação dos atributos das referências. A padronização pode ser executada por algoritmos computacionais, porém não tão efetivos quanto aos resultados produzidos por pessoas (CHIANG *et al.*, 2008, WU *et al.*, 2007).

Dois padrões comuns adotados para inserir um valor de atributo em um registro são o formato de campo fixo e o delimitador de caractere. No formato de campo fixo cada valor de atributo é um campo do registro que tem sempre a mesma posição inicial e final. Em formatos delimitados por caracteres, os valores dos atributos estão em uma lista ordenada separados por um caractere especial chamado delimitador de campo.

Vírgulas e caracteres de tabulação são comumente usados para esse fim e, por essa razão os arquivos no formato delimitado por caracteres são chamados de *Comma-Separated Values* (CSV).

Outro padrão amplamente utilizado é o XML (*eXtensible Markup Language*) (EXTENSIBLE MARKUP LANGUAGE, 2014) que representa um formato estruturado em que os valores dos atributos inseridos estão entre *tags*.

Na Tabela 2, é um exemplo de como o nome "JOHN SMITH" está representado em três formatos diferentes. A primeira como posição fixa, a segunda como delimitado pelo caractere "“" e a terceira utilizando XML.

O interesse na atividade ERA1 tem crescido junto com a percepção de que uma grande quantidade de informações úteis de uma organização muitas vezes reside em formatos não estruturados. INMON & NESAVICH (2007) sugerem que na maior parte das vezes, a maioria das informações de uma organização existe em formatos não estruturados (*Unstructured Textual Information* – UTI).

Tabela 2 – JOHN SMITH em três formatos estruturados (adaptado de TALBURT (2010))

Char position	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
Fixed position	J	O	H	N						S	M	I	T	H	,	J	R		
Delimited	“	J	O	H	N	“	,	“	S	M	I	T	H	,	J	R	“		
XML	<	D	O	C	>	<	L	>	S	M	I	T	H	,	J	R	<	/	L
XML (cont)	>	<	F	>	J	O	H	N	<	/	L	>	<	/	D	O	C	>	

3.4.2.ERA 2 – Preparação de Referência de Entidade

Mesmo quando referências a entidades estão em um formato estruturado, é necessário um pré-processamento das fontes de referência de entidade antes do processo de resolução efetiva ocorrer. Coletivamente, esse pré processo é chamado ETL (*Extract Transform Load*), onde ETL é um acrônimo para extração, transformação e carga. Este processo, também é chamado de limpeza de dados. A seguir estão as operações comumente usadas para preparar referências para o processamento de ER (CHAN *et al.*, 2009):

- *Encoding* – Converter *encode* de um dado para outro; Por exemplo, UTF-8 para ASCII;
- Conversão – Transformar a representação de um formato de dado para outro. Por exemplo, um dado do tipo inteiro para texto;
- Padronização – Transformar a representação de dados para um padrão previamente definido. Por exemplo, alterar o texto "Logradouro" e "Logr" para "LOG";
- Correção – Alterar valores baseados em base de dados confiáveis. Por exemplo, corrigir um código de CEP a partir do nome do logradouro, município e UF pelos Correios[28].
- Validação – Criar regras lógicas baseados nos valores dos dados. Por exemplo, verificar se uma compra foi realizada, anteriormente ao cadastro do cliente;
- Melhoria – Acrescentar informações que não estão contidas na referência original. Por exemplo, adicionar coordenadas de latitude e longitude baseado nos dados do endereço da rua.

Nesta fase é que as áreas de ER e IQ (*Information Quality*) estão intimamente relacionadas, pois a melhora dos dados aumenta as chances de sucesso da resolução de entidades. E consequentemente, a resolução de entidades contribui para a melhora geral da qualidade da informação no sistema (TALBURT, 2010).

3.4.3.ERA 3 – Resolução de Referência de Entidade

A etapa ERA3 é onde são tomadas as decisões sobre a equivalência de duas instâncias de referência. Importante salientar que dois termos normalmente se confundem nessa etapa: o *matching* e o *linking*. Enquanto o primeiro calcula o grau de similaridade entre os valores dos atributos de identidade em duas referências, o segundo termo agrupa referências equivalentes.

É muito comum as referências equivalentes terem *link value* e não terem *matching* dos atributos de identidade. TALBURT (2010) dá como exemplo uma cliente chamada Mary Jone que mora na Rua Oak e que se casa com John Smith. Mary tem

[28] http://www.buscacep.correios.com.br

seu nome alterado para Mary Smith e posteriormente eles mudam para a Rua Elm. Neste caso, as referências Mary Jones da Rua Oak e Mary Smith da Rua Elm apresentam diferentes sobrenomes e endereços, entretanto representam a mesma entidade.

Outra confusão com terminologia, é o termo *duplicate*, *duplicate records* ou *record deduplication* (TALBURT, 2010). O autor afirma que é senso que o termo *duplication* implica no nível máximo de similaridade, indicando que uma referência é cópia exata da outra. Entretanto, em outros contextos, o termo pode ser usado para indicar uma similaridade parcial e ainda em outros contextos como referência equivalente (NAUMANN & HERSCHEL, 2010).

ER trata de tomar decisões e *linking* é uma maneira de caracterizar esta decisão (Princípio 2 de ER) (TALBURT, 2010). Embora o conhecimento em que duas referências tenham similaridade alta em alguns atributos ajude no processo de decisão, o *matching* não é um fator determinante.

Em relação ao que foi discutido, TALBURT (2010) criou uma lei fundamental de ER: "Duas referências de entidades devem ser ligadas se, e somente se, eles são equivalentes (referenciam a mesma entidade)[29]."

3.4.3.1.Referências correspondentes e equivalentes

A fim de exemplificar os conceitos de referência correspondentes e equivalentes, TALBURT (2010) constrói um cenário em que existem três conjuntos.

O primeiro, conjunto S, é representado pelas referências das entidades. A notação SxS representa um produto cartesiano de S sobre si mesmo, isto é, um conjunto de todos os pares de referências em S. Um algoritmo pode facilmente calcular o grau de similaridade de cada par e definir se as referências são iguais ou não de acordo com um critério definido. O segundo, conjunto M, que denota um subconjunto de SxS e que contém todos os pares correspondentes, isto é, o conjunto M representa os possíveis pares equivalentes. E o terceiro, um conjunto E, que denota um subconjunto de SxS e que contém todos os pares equivalentes.

[29] Tradução do autor para: "Two entity references should be linked if and only if they are equivalent (reference the same entity)."

Resumindo, o conjunto E representa os resultados corretos na resolução de entidade, enquanto M representa os resultados correspondentes. O grau de intercessão entre os dois conjuntos mostra que o processo de *matching* utilizado foi efetivo ou não. Quanto menores são os conjuntos E-M e M-E, mais efetivo é o algoritmo de *matching* para determinar a equivalência.

A Figura 3 mostra graficamente a intercessão desses dois subconjuntos.

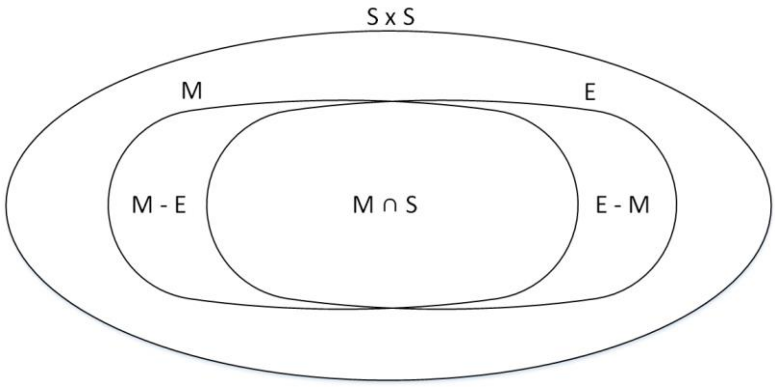


Figura 3 – Referências correspondentes e equivalentes (adaptado de TALBURT (2010))

A Figura 3 pode ser interpretada de uma maneira mais geral. Suponha que M é considerado um processo de decisão que diz que sim quando dois registros são equivalentes, e não quando eles não são. O conjunto E ainda mantém como os resultados corretos. Algumas considerações podem ser tomadas:

- M-E é o conjunto dos resultados falso positivos. Positivo indica que o processo de decisão foi sim porque está contido no conjunto M, e falso porque o conjunto não está contido em E, o que significa que eles não são equivalentes;
- E-M é o conjunto de resultados falso negativos. Negativo porque o processo de decisão foi não (não está contido em M), porém é uma decisão falsa, pois os pares estão no conjunto E. Isto é, deveriam ser equivalentes;
- M∩E é o conjunto de resultados verdadeiro positivos;
- SxS - (M∪E) é o conjunto de verdadeiro negativos.

O verdadeiro positivo e o verdadeiro negativo são as decisões corretas, o restante são decisões incorretas. A Tabela 3 resume de forma clara essas decisões.

Tabela 3 – Classificação do Processo de Decisão de Resultados (adaptado de TALBURT (2010))

	Decisão que deveria ser Sim	Decisão que deveria ser Não
Decisão em Sim	Verdadeiro Positivo	Falso Positivo
Decisão em Não	Falso Negativo	Verdadeiro Negativo

3.4.3.2.Problema dos Falsos Negativos

Os falsos negativos são um problema constante na maioria dos contextos de ER. A razão para isso é que a avaliação do processo de ER é focada sobre decisões positivas e não negativas.

TALBURT (2010) explica o problema de forma clara ao citar que, em geral, o conjunto das decisões negativas é muito maior do que as positivas. O autor ainda exemplifica um caso em que o conjunto S possui 10.000 referências, logo SxS tem 100 milhões de pares. Se o processo de decisão para os pares positivos for de 1%, então 99 milhões são decisões negativas. Portanto, o conjunto E-M (falsos negativos) é uma porção muito menor do que (SxS)- M (decisões negativas) (Princípio 3 de ER) TALBURT (2010).

A melhor técnica para detectar pares de falsos negativos é por amostragem de pares com decisão negativa (ARKADY, 2007). TALBURT (2010) ratifica ainda que, o impacto de uma decisão falsa negativa é muito menor do que uma decisão falsa positiva, pois afirmar que duas referências são equivalentes, sendo que não são, pode trazer problemas futuros quanto unir registros de duas pessoas distintas como se fosse a mesma, por exemplo. Caso duas referências sejam equivalentes pelo processo de decisão, seus dados podem ser unificados de modo a consolidar as informações. Entretanto, caso esses pares sejam falsos positivos, será problemático, pois na realidade se estará fundindo duas referências não equivalentes, comprometendo a consistência dos dados (Princípio 4 de ER) TALBURT (2010).

3.4.4.ERA 4 – Gerenciamento de Identidade de Entidade

Outro importante termo relacionado à resolução de entidades é a identidade de entidade[30]. LIM *et al.*(1993) definem este termo como um conjunto de valores de atributos para entidade com um conjunto de regras de distinção, que permitem que a entidade seja diferenciada de todas as outras entidades da mesma classe em um contexto específico. A partir desta definição, a entidade de resolução pode ser realizada através da comparação desses atributos de identidade. Existem diversas razões para que o método de *direct matching* esteja longe da solução perfeita em resolução de entidades (TALBURT, 2010). Logo abaixo seguem alguns exemplos:

- O aumento de entidades em relação ao contexto original. Os valores de regras e atributos criados para discernir 100 entidades diferentes podem não ser suficientes para diferenciar caso sejam adicionadas mais 200 entidades.
- Variação ou erro na representação de valores de atributos, erros de ortografia, valores nulos e valores padrão.
- Nem todos os atributos estão representados em cada referência. Referências geralmente representam apenas uma projeção do conjunto total de atributos de identidade. Por isso, em um caso onde houvessem duas referências equivalentes, porém, sem nenhum *matching*, seria dado como não equivalentes. Entretanto, se um atributo, como CPF, fizesse parte das referências, claramente seriam identificadas como equivalentes.

Em ambientes onde existem pequenos números de registros, é possível identificar uma entidade com poucos registros. Possivelmente, uma empresa de pequeno porte pode identificar unicamente cada funcionário apenas pelo nome e sobrenome. Entretanto, em uma população extremamente maior, dificilmente uma entidade pode ser representada desta maneira. No Brasil, é muito comum as pessoas terem o mesmo nome, nome do meio e sobrenome. Até mesmo esses dados com a data de aniversário podem não ser suficientes para identificar uma única pessoa.

[30] Tradução do autor para: "Entity identity"

A resolução de identidade[31] é um processo de ER em que as referências são resolvidas através da comparação de um conjunto pré-conhecido de identidades. A resolução ocorre entre duas referências de entrada, e uma delas deve pertencer necessariamente ao conjunto pré-conhecido. Um exemplo prático na utilização de resolução de identidade é um sistema de reconhecimento de clientes que determina se um cliente existe ou não baseado em um banco de dados. Ferramentas que possibilitam a consulta em tempo real são extremamente valorizadas no mercado.

Caso pelo menos uma das referências da entidade faça parte do conjunto já conhecido, a identificação da entidade pode ser uma técnica para executar a resolução de entidades. No caso em que as duas referências sejam conhecidas do sistema, claramente os resultados de resolução já são conhecidos também. E se apenas uma entidade é conhecida e a outra não, as referências têm diferentes identidades. Existe apenas um caso em que a resolução de entidade não consegue realizar a identificação, que é quando as duas referências são entidades externas ao conjunto de entidades conhecidas.

TALBURT (2010) ilustra um exemplo da diferença entre a resolução de entidades e resolução de identidade. Dado um cenário onde existem duas impressões digitais, estas, podem ser comparadas uma a outra e ditas equivalentes ou não (resolução de entidades). Caso se queira identificar as digitais, será necessário um banco de dados onde as digitais suspeitas já estejam previamente cadastradas (resolução de identidade).

Quando a resolução de identidade é definida como parte do processo de resolução de entidades de referência com uma entidade conhecida, indica que que a resolução de identidade e resolução de entidades não são o mesmo processo. Ainda mais, a resolução de identidade é apenas uma forma de resolução de entidades (Princípio 6 de ER) TALBURT (2010).

A identidade desempenha um papel importante nos sistemas de ER. Compreender o papel da identidade na ER indica a definição anterior, que é a identidade de entidade como sendo um conjunto de regras e atributos que distinguem entre as entidades em um dado contexto, e com o Princípio ERA 1, que estabelece que os

[31] Tradução do autor para: "Identity resolution"

sistemas de ER trabalham com referências a entidades, não as entidades propriamente ditas.

Entidades reais têm diversos números de atributos. Uma pessoa pode ser descrita (atributos) de inúmeras maneiras, como: altura, peso, cor dos olhos, formação acadêmica, data de nascimento e cidade natal, por exemplo. No entanto, o modelador do sistema escolherá somente alguns atributos para representar a identidade dessa entidade no sistema. Outros modeladores poderão escolher outras características da entidade pessoa. O maior problema na perspectiva de ER é quando diferentes sistemas geram referência para a mesma entidade usando diferentes atributos de identidade.

Um exemplo de múltiplas referências em sistemas diferentes são as redes sociais. Muitas pessoas têm diversas "identidades online" e ligá-las está se mostrando uma tarefa bem complexa. Trabalhos relacionados à aplicação de resolução de entidades nessa área começaram a ser aplicados em 2006 (BILGIC *et al.*, 2006).

3.4.4.1. Visualização de Identidades Internas e Externas

TALBURT (2010) ilustrou a questão da visualização de identidades com um exemplo de histórico de mudanças de uma pessoa. A Figura 4 apresenta este exemplo através de uma pessoa chamada Mary Smith, que, a partir de março de 2000, trocou o seu nome para Mary Jones e seu endereço, e, em novembro de 2011, se mudou para outro endereço. A pessoa chamada Mary é a mesma no mundo real para todos os três históricos.

Uma maneira de descrever esta situação é, em termos de uma visão interna contra uma visão externa de identidade (TALBURT *et al.*, 2009).

A visão interna da situação apresentada pela Figura 4 seria do ponto de vista da própria pessoa, ou algum parente bem próximo, alguém que tenha um contato muito próximo sobre sua vida e histórico, isto é, a visão interna da identidade representa um modelo de universo fechado em que para um dado conjunto de atributos de identidade, todos os valores de atributo são conhecidos para o visualizador interno, e qualquer valor para um destes atributos que não é um valor conhecido pertence à uma identidade diferente (TALBURT, 2010).

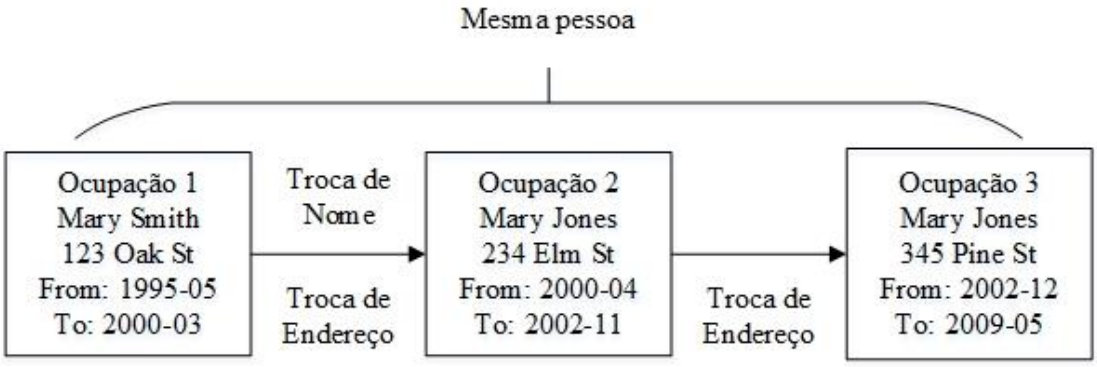


Figura 4 – Histórico de Ocupações (adaptado de TALBURT (2010))

A segunda visão, a externa de identidade, é aquela em que foi coletado um número de valores de atributos de uma identidade, mas sua corretude ou completude são desconhecidas. Quando um sistema com base em visão externa recebe uma referência, o sistema deve processar e escolher ligá-la a uma identidade existente ou se deve criar uma nova. Portanto, uma visão externa representa um modelo de universo aberto, porque ao contrário da visão interna, o sistema não pode assumir que ele tem conhecimento completo de todos os valores de identidade.

Um fato que pode ocorrer, por exemplo, é que a Ocupação 3 não esteja associada às outras duas ocupações. Isso torna a Ocupação 3 outra identidade, diferente das outras. Consequentemente, devido a esse problema de não ter os dados completos, o sistema pode construir uma visão imprecisa da identidade. Quando apresentado um novo registro de ocupação, o sistema pode associá-lo erroneamente à uma identidade inexistente ao qual não pertence. TALBURT (2010) comenta novamente a relação estreita entre ER e IQ (*Information Quality*), pois acurácia e completude são duas das 16 dimensões de informação da qualidade descritos pelo *Framework* de Wang-Strong (WANG & STRONG, 1996).

TALBURT (2010), afirma que um sistema ER com base em visão externa constrói seu conhecimento sobre a identidade das entidades pouco a pouco, ou seja, de forma incremental. O autor ainda comenta que a visão externa de identidade se assemelha, em muito, aos esforços de uma empresa ou governo que tem de unificar a sua visão de clientes.

Em geral, todos os sistemas de ER usam a identidade da referência em algum nível para a resolução de entidades, porém nem todos os sistemas possuem um gerenciamento de identidade. Um dos processos mais básicos é o *merge-purge*, que utiliza a identidade realizando *matching* em seus atributos, e considera os atributos com

valores próximos equivalentes e posteriormente realiza a combinação. Sistemas que têm como base o *merge-purge* não gerenciam a identidade das entidades, pois esse conhecimento é recriado a cada vez que o processo é executado, uma vez que os dados são descartados ao final do processo (TALBURT, 2010).

O gerenciamento de identidade em um sistema de ER ocorre quando são carregadas informações de identidade, ou quando se mantém a totalidade ou parte da informação de identidade da entidade a partir das referências que foram resolvidas. Sistemas de ER que suportam o gerenciamento de identidade têm uma ampla vantagem sobre outros sistemas, como o armazenamento das persistências de links entre as referências, possibilidade de processamento transacional e aplicação de técnicas de associação e declaração. (Princípio 7 de ER) TALBURT (2010).

3.4.5.ERA 5 – Análise de Relação de Entidade

Após as referências equivalentes terem sido agrupadas (ERA3) e possivelmente identificadas (ERA4), a próxima etapa proposta por muitas ferramentas é encontrar relacionamentos que essas referências equivalentes têm em relação ao restante da base de dados.

Household relationship foi o primeiro conceito a explorar as diversas relações que as referências de entidades do tipo clientes possuem. Criado por empresas de marketing, este conceito visa procurar entender as relações de moradias de familiares. Outra forma de explicar o conceito seria que "... são todas as pessoas que residem no mesmo endereço e que têm o mesmo sobrenome."[32] (TALBURT, 2010).

TALBURT (2010) alega que embora o conceito seja simples, ele não capta as exceções que existem na sociedade, como mulheres divorciadas com nome de casada, ou mulheres casadas que não alteraram o seu nome. Logo, a relação de entidade não é tão simples quanto parece. Referências podem estar relacionadas de formas que não esteja clara esta relação.

Segundo WATTS & STROGATZ (1998) o relacionamento entre as entidades pode ser categorizado em níveis de separação. O ERA3, nesse contexto, seria uma atividade que busca duas referências com nível de separação zero, o que significaria

[32] Tradução do autor para: "[...] all the people at the same address with the same last name."

referências equivalentes. Grau um de separação seria uma associação direta entre duas referências como o *household relationship*, Para grau dois de separação seria uma associação intermediária ou transitiva. Por exemplo, suponha que duas pessoas A e B residam na mesma casa, e as pessoas B e C frequentem o mesmo clube (Figura 5). O grau de separação entre A e B, e B e C é um; já o grau entre A e C é dois (TALBURT, 2010).

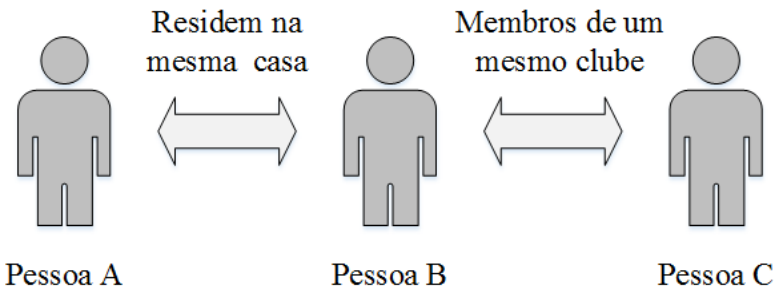


Figura 5 – Relação entre referências (adaptado de TALBURT (2010))

3.5.Técnicas de *Matching*

As técnicas de matching (ou casamento) são largamente utilizados na resolução de entidades aplicados a algoritmos computacionais. A seguir, são apresentadas alguns dos algoritmos mais comuns utilizados tanto no meio acadêmico ou comercial.

3.5.1.Casamento Direto (Direct Matching)

Correspondência direta é o cálculo do grau de similaridade entre os atributos de identidade de duas referências. Existem basicamente cinco métodos: (1) Correspondência exata, (2) Diferença Numérica, (3) Correspondência Sintática Aproximada, (4) Correspondência Semântica aproximada e (5) Código de Correspondência Derivada.

A correspondência exata é a forma mais simples de comparação, pois o resultado só pode ser verdadeiro ou falso. O problema dessa categoria na comparação de textos se deve à baixa probabilidade de um texto ser exatamente idêntico ao outro. Logo, caso os textos tenham variação de um caractere, ou mesmo esteja em maiúsculo e outro em minúsculo, o retorno será falso. O sucesso para esta categoria é uma boa

etapa ERA2 aplicada às referências, pois na ERA 2 é onde são aplicados os métodos de limpeza dos dados.

A comparação entre dois campos de texto pode ser realizada sintaticamente ou semanticamente. Os algoritmos ASM (*Approximate String Matching*) são utilizados em comparações sintáticas (HERZOG *et al.*, 2007).

3.5.1.1.Distância de Levenshtein

Este foi um dos primeiros algoritmos a surgir e que busca mensurar a similaridade através do número mínimo de operações que uma *string* deve passar para se tornar outra (LEVENSHTEIN, 1966). Normalmente as operações são: inserir um caractere, apagar um caractere, substituir um caractere, ou também em algumas variações do algoritmo, transpor dois caracteres adjacentes.

3.5.1.2.Q-Gram

O algoritmo q-gram leva em conta a ordenação dos caracteres. Um q-gram é uma sequência fixa de caracteres de tamanho q (UKKONEN, 1992).

O princípio da similaridade de q-gram entre duas *strings* é que quanto maiores forem os trechos que derivam da primeira *string* e são encontrados na segunda *string*, mais similares elas serão.

Tabela 4 – Cálculo de q-gram (adaptado de TALBURT (2010))

	1-gram	2-gram	3-gram
JULIE	"J", "U", "L", "E"	"JU", "UL"	"JUL"
JULIES			

Considere duas *strings* "JULIE" e "JULES" apresentadas na combinação de q-grams na Tabela 4. Nesse exemplo, o valor máximo de q foi 3, quando dividido pelo tamanho da *string* de 5, dando um grau de similaridade de 60%.

3.5.1.3.Jaro

Uma variação do q-gram é o Comparador de *String* de Jaro (JARO, 1989). Ele considera o número de caracteres em comum entre duas *strings* e o número de caracteres com transposições. A Figura 6 mostra a fórmula de similaridade de Jaro.

$$J(A,B) = W_1.\frac{C}{L_A} + W_2.\frac{C}{L_B} + W_3.\frac{(C-T)}{C}$$

Figura 6 – Fórmula para Comparador de String de Jaro (adaptado de JARO (1989))

A e B representam duas *strings* que serão comparadas. W1, W2 e W3 são pesos definidos para a primeira *string*, segunda *string* e as transposições, respectivamente. O somatório de W1 + W2 + W3 deverá ser 1. C é a quantidade de caracteres em comum entre as duas *strings*. T é o número de transposições. LA e LB são os tamanhos das *strings* A e B, respectivamente.

3.5.1.4.Jaro-Winkler

A Distância de Jaro-Winkler é uma métrica de distância entre *strings* que é uma variação da distância do comparador de Jaro (JARO, 1989). Dadas duas *strings* s1 e s2, *m* sendo o número de caracteres semelhantes e *t* o número de transposições, o cálculo da distância de Jaro-Winkler é representado pela fórmula: *1/3(m/|s1| + m/|s2| + (m – t)/m).*

Alguns algoritmos não só exigem que a sequência de caracteres seja o mesmo, mas levam em conta a posição em que essa sequência se inicia. Estes algoritmos q-gram são chamados posicionais. O algoritmo de Jaro-Winkler (WINKLER, 1999) é um algoritmo q-gram posicional. Ele é uma modificação do tradicional algoritmo de Jaro (1989), porém com pesos adicionais sobre os quatro primeiros caracteres das duas *strings*. Outra fórmula para a distância de Jaro-Winkler com base no algoritmo de Jaro está representada na Figura 7.

$$W(A,B) = J(A,B) + 0.1 * N * \big(1 - J(A,B)\big)$$

Figura 7 – Fórmula de Distância de Jaro-Winkler (adaptado de TALBURT (2010))

3.5.1.5.Jaccard

Similaridade de Jaccard (JACCARD, 1901) é uma métrica de similaridade entre conjuntos, comumente utilizada para calcular a distância entre palavras/frases. Esta

métrica de distância está sendo usada amplamente em algoritmos aplicados na área de resolução de entidades para calcular a distância entre registros, como no projeto CrowdER (WANG *et al.*, 2012). Para se calcular a Similaridade de Jaccard, é preciso dividir o módulo do conjunto interseção dos conjuntos pelo módulo do conjunto união (Figura 8).

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|}$$

Figura 8 – Fórmula de similaridade de Jaccard (adaptado JACCARD (1901))

3.6.Sistemas ER

Nesta seção serão apresentas algumas das ferramentas mais comuns e importantes no mercado de resolução de entidades. Os sistemas têm diferentes enfoques tratando diferentes atividades de ER. Algumas são utilizadas até pelo governo americano para a identificação de pessoas, como será visto abaixo.

3.6.1.DataFlux dfPowerStudio

DataFlux® foi fundada em 1997 e após três anos foi adquirida pela SAS Institute® e ao longo desse tempo evoluiu para fornecer uma ampla gama de aplicações, incluindo a integração de dados, MDM e qualidade dos dados. Nos últimos anos, a empresa tem posição de destaque na Magic Quadrant® for Data Quality Tools, publicado pela Research Group® Gartner (SAS, 2014).

A ferramenta dfPowerStudio® é utilizado para executar a técnica de *merge-purge* de dados de clientes que envolvem as atividades de preparação de referência (ERA2) e etapas de resolução (ERA3).

3.6.2.Infoglid Identity Resolution Engine

Fundada em 1996, a *Infoglide Software*®, uma empresa privada, e tem como foco o desenvolvimento e comercialização de software de resolução de identidades para o mercado comercial e governamental. Sendo uma das principais empresas nesse tipo de ramo, os seus sistemas apoiam alguns sistemas do departamento TSA (*Transportation Security Administration*) dos Estados Unidos (INFOGLIDE SOFTWARE, 2014).

O carro-chefe da empresa é a ferramenta *Identity Resolution Engine*® (IRE). O IRE é baseado na arquitetura de unificação de base de dados heterogêneas, onde o *engine* funciona como um *hub* que conecta e consulta múltiplas base de dados e tabelas. Os resultados são combinados e apresentados para o usuário. Embora o IRE possa operar de forma interativa ou em lote, ele é especializado em explorar e encontrar relações não óbvias nos relacionamentos entre as entidades, tarefa executada no ERA 5 (TALBURT, 2010).

Dentre outros serviços do IRE, estão: Resolução de Identidades, Busca de Similaridades, *Link* de Descoberta Social e Resolução de Anonimato para Dados Privados.

3.6.3.OYSTER

OYSTER é um projeto de desenvolvimento de software de código aberto patrocinado pelo Centro de Pesquisa na Universidade de Arkansas em Little Rock[33]. OYSTER (*Open sYSTem Entity Resolution*) é um sistema de resolução de entidade que pode ser configurado para ser executado em vários modos de operação, incluindo *merge-purge*, captura de identidade e resolução de identidade. O motor de resolução suporta correspondência direta probabilística, equivalência transitiva, e equivalência assertiva (TALBURT, 2010).

A versão original de OYSTER foi projetado para apoiar a resolução de entidade para registros de alunos, porém o sistema pode facilmente processar uma gama de domínios de tipos de entidades.

O sistema OYSTER realiza também o gerenciamento de identidade e também suporta identificadores persistentes de identidade. O sistema foi desenvolvido em Java e seu código-fonte e documentação estão disponíveis[34] para uso sob a licença de código aberto. As informações são processadas em tempo real por meio de scripts XML definidos pelo usuário.

[33] http://ualr.edu/eriq

[34] http://ualr.edu/eriq/downloads

3.7.Métricas de ER

As métricas são necessárias para avaliar, se de fato, um algoritmo ou modelo são eficientes em suas abordagens. Algumas métricas são utilizadas em contexto gerais como o *recall* e *precision*. Enquanto outras métricas, como TWI foi criado com o propósito de avaliar de forma simplificada o grau de similaridade entre dois conjuntos produzidos, em geral, por algoritmos aplicados a resolução de entidades. A seguir, são apresentadas algumas das técnicas mais comuns na área de resolução de entidades, e que serão utilizadas nos capítulos posteriores para mostrar que o modelo proposto é viável em termos de eficiência.

3.7.1.Talburt-Wang Index (TWI)

Uma das métricas que foi desenvolvida para comparar resultados de ER e de fácil aplicação computacional é o Talburt-Wang Index (TWI) (TALBURT *et al.*, 2007).

O TWI consiste em calcular o grau de similaridade entre duas partições (ou agrupamentos). O resultado do valor sempre será um valor entre 0 e 1; quanto maior o valor, maior é o grau de similaridade.

Dada uma partição A e outra B de um conjunto S; e sendo V (Figura 9) o conjunto de todas as interseções não nulas entre as partições A e B; então TWI será definido como a Figura 10.

$$V = \{A_i \cap B_j | A_i \in A, B_j \in B, and\ A_i \cap B_j \neq \emptyset\}$$

Figura 9 – Fórmula para o cálculo do conjunto V (adaptado de TALBURT (2010))

Caso A seja considerado o grupo das relações corretas, e B um grupo resultante de uma aplicação de algoritmos computacionais, então o TWI pode ser interpretado como a acurácia de resolução de entidade (ZHOU & TALBURT, 2011).

$$TWI = \frac{\sqrt{|A| \,.\, |B|}}{|V|}$$

Figura 10 – Fórmula para o cálculo de TWI (adaptado de TALBURT *et al.*(2007))

3.7.1.1.Cálculo passo a passo de TWI

Segundo a Definição 3.4 do livro de TALBURT (2010), um cenário ER é uma tripla (E, S, ω), sendo E um processo de resolução de entidade, S um conjunto de referências de entidades e ω representa a sequência em que as referências serão processadas.

A fim de ilustrar o cálculo de similaridade do T-W Index, o processo E foi definido assim como o conjunto S foi fixado, só alterando a sequência para serem geradas diferentes partições.

A coluna "SCode" das Tabela 5 e Tabela 6 representa o *linking* para agrupar referências equivalentes geradas pelos algoritmos.

As duas partições A e B representam os resultados dos cenários (E, S, ω1) e (E, S, ω2) respectivamente, e o conjunto V representa a interseção das partições A e B.

A partição "A" possui duas subpartições {{r1, r3, r5}, {r2, r4}} e

Tabela 5 – O resultado de um cenário ER (E, S, ω1) (adaptado de TALBURT (2010))

	S, ω1				**Output (E, S, ω1)**				
	First	Last	DOB	SCode	Ident	First	Last	DOB	SCode
r1	Edgar	Jones	20001104	G34	1	Edgar	Jones	20001104	G34
r2	Mary	Smith	19990921	G55	2	Mary	Smith	19990921	G55
r3	Eddie	Jones	20001104	H15	3	Eddie	Jones	20001104	G34
r4	Mary	Smith	19990921	H17	4	Mary	Smith	19990921	H17
r5	Eddie	Jones	20001104	G34	5	Eddie	Jones	20001104	H15

A partição "B" possui {{r1, r3}, {r2, r4}, {r5}}, três subpartições.

Tabela 6 – O resultado de um cenário ER (E, S, ω2) (adaptado de TALBURT (2010))

	S, ω1				**Output (E, S, ω2)**				
	First	Last	DOB	SCode	Ident	First	Last	DOB	SCode
r1	Edgar	Jones	20001104	G34	1	Edgar	Jones	20001104	G34
r2	Mary	Smith	19990921	G55	2	Mary	Smith	19990921	G55
r3	Eddie	Jones	20001104	H15	3	Eddie	Jones	20001104	H15

r4	Mary	Smith	19990921	H17	4	Mary	Smith	19990921	H17
r5	Eddie	Jones	20001104	G34	5	Eddie	Jones	20001104	G34

A Tabela 7 representa o cálculo da interseção entre o conjunto A e B, tendo como resultado final o valor 3.

Tabela 7 – Interseção da Matriz de (E, S, ω1) e (E, S, ω2) (adaptado de TALBURT (2010))

A/B	{r1, r3}	{r2, r4}	{r5}	
{r1, r3, r5}	2	0	1	3
{r2, r4}	0	2	0	2
	2	2	1	5

E finalmente, Figura 11 representa o cálculo do T-W Index. Caso a partição "A" fosse o gabarito das referências, e a partição "B" gerada pela multidão, a acurácia da multidão seria de 81,6%.

$$TWI = \frac{\sqrt{|A| \,.\, |B|}}{|V|} = \frac{\sqrt{2 \,.\, 3}}{3} = \frac{2.494}{3} = 0.816$$

Figura 11 – Cálculo de similaridade (adaptado de TALBURT (2010))

3.7.2.Recall e precision

Recall e *precision* são medidas básicas usadas na avaliação de estratégias de busca. *Recall* é a razão entre o número de registros relevantes recuperados e o número total de registros relevantes na base de dados. Por exemplo, em uma busca de texto em um conjunto de documentos, o *recall* é dado como a divisão entre o número de resultados corretos e o número de resultados que deveriam retornar.

Em contrapartida, *precision* é a razão entre o número de registros relevantes encontrados e o número de registros relevantes e irrelevantes recuperados. Em uma busca de texto em um conjunto de documentos, por exemplo, a *precisão* é dada como o número de resultados corretos divididos pelo total de resultados.

Recall e *precision* são expressos em percentagem. Esses são geralmente objetivos conflitantes, considerando que se um quer ver mais itens relevantes, geralmente mais itens irrelevantes também são recuperados.

Dado um conjunto A que representa os dados corretos, e o conjunto B que são os dados a serem avaliados, a Figura 12 representa a fórmula para o cálculo do *recall* e *precision*.

$$recall = \frac{A \cap B}{A} \quad e \quad precision = \frac{A \cap B}{B}$$

Figura 12 – Cálculo de *recall* e *precision* (elaborado pelo autor)

A Figura 13 ilustra a relação entre o conceito de *recall* e *precision* em forma de conjuntos.

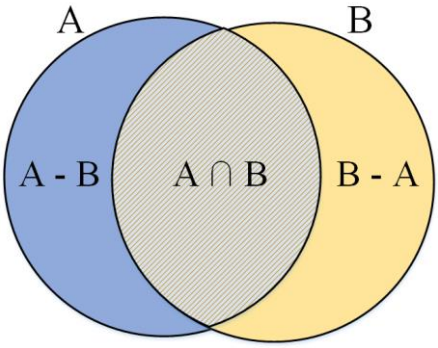


Figura 13 – Representação de conjuntos para cálculo de *recall* e *precision* (elaborado pelo autor)

3.7.3.Medida Pairwise e Cluster-level

MENESTRINA *et al.* (2005) levantam a discussão da comparação "*pairwise*" em relação a "*cluster-level*", e propuseram uma nova medida chamada *merge distance*. A medida *pairwise* é similar à Medida *Rand Index*, em que são contados pares no interior das classes de partições (clusters). No entanto, no caso da medida *pairwise*, os pares distintos são apenas contados.

Dada duas partições A = {{r1, r3, r5},{r2, r4}} e B = {{r1, r3},{r2, r4},{r5}}, a partição A gera um conjunto de quatro pares distintos: Pair (A) = {(r1, r3), (r1, r5), (r3, r5), (r2, r4)}; e a partição B um conjunto de 2 pares: Pair (B) = {(r1, r3), (r2, r4)}.

Ainda, dadas as partições A e B, TALBURT (2010) define *Pair precision* e *Pair recall* como apresentados nas Figura 14 e Figura 15.

$$PairPrecision\ (A,B) = \frac{|Pairs(A) \cap Pairs\ (B)|}{|Pairs(A)|}$$

Figura 14 – Fórmula de *Pair Precision* (adaptado de TALBURT (2010))

$$PairRecall\ (A,B) = \frac{|Pairs(A) \cap Pairs\ (B)|}{|Pairs(B)|}$$

Figura 15 – Fórmula de *Pair Recall* (adaptado de TALBURT (2010))

3.7.4.F-Score

Para avaliar a o desempenho da recuperação dos dados, é possível a utilização de uma medida chamada F-Score. Esta medida é dada como a média harmônica (MITCHELL, 2004) da precisão e do *recall*. A versão universal do F-Score emprega um coeficiente β, através do qual a razão *precision-recall* pode ser personalizada. Abaixo segue a fórmula básica de F com coeficiente β = 1 (Figura 16):

$$F_\beta = \frac{(1+\beta^2)\cdot P \cdot R}{\beta^2 P + R}, F = F_1 = 2\frac{P \cdot R}{P + R}$$

Figura 16 – Fórmula F-Score descrita em função de *Precision*, *Recall* e coeficiente *Beta* (adaptado de GOUTTE & GAUSSIER (2005))

O coeficiente β permite que *precision* ou *recall* tenham maior relevância, e ambos estão balanceados quando β = 1. Sabe-se que a *precision* e o *recall* podem ser calculados com base nos falso-positivos e negativos e nos verdadeiro-positivos, como segue na Figura 17, onde TP é o número de verdadeiro-positivos, FP, o número de falso-positivos, e FN, falso-negativos.

$$P = \frac{TP}{TP + FP}, R = \frac{TP}{TP + FN}$$

Figura 17 – Fórmula de *Precision* e *Recall* descrita em falso-positivos, falso-negativos e verdadeiro-positivos (adaptado de GOUTTE & GAUSSIER (2005))

Dessa forma, a fórmula F-Score, também chamado F-measure, pode ser escrita na forma apresentada na Figura 18.

$$F_\beta = \frac{(1+\beta^2)TP}{(1+\beta^2)TP + \beta^2 FN + FP}$$

Figura 18 – Fórmula F-Score descrita em função de falso-positivos, falso-negativos, verdadeiro-positivos e coeficiente *beta* (adaptado de GOUTTE & GAUSSIER (2005))

Outras duas formas comuns de medidas F-Score são F_2, o qual dá maior peso para a *recall*, e $F_{0.5}$, o qual dá maior ênfase ao *precision*.

3.7.5.Eficácia

A medida da eficácia pressupõe que, para uma coleção de referências M, o número verdadeiro de entidades a que se referem é conhecido. A eficácia é uma média harmônica da precisão e do *recall* de cada entidade (TANSEL et al., 2006). É indicado para contextos que lidam com resolução de identidades (ERA5), pois o número *n* (número de entidades do mundo real) já é conhecido de antemão. A fórmula da eficácia é apresentada na Figura 19.

$$\frac{1}{N}\left(\sum_{i=1}^{n} \frac{2 \ * \ \text{precision(i)} \ * \ \text{recall(i)}}{\text{precision(i)} + \ \text{recall(i)}}\right)$$

Figura 19 – Fórmula de Eficácia (adaptado de TANSEL *et al.* (2006))

3.8.Resolução de Entidades utilizando a multidão

A resolução de entidade é um processo originalmente construído a partir de algoritmos que são limitados, pois a variedade de dados e fontes faz a comparação de entidades ser um verdadeiro caos.

3.8.1.A multidão

A multidão tem um enorme potencial para a contribuição na resolução de entidades. Isto se deve à capacidade de comparação e interpretação que não está presente em nenhum algoritmo existente.

Algoritmos se prendem a regras pré-estabelecidas, enquanto a multidão tem seu próprio modo de pensar e raciocinar. É com esse diferencial que os sistemas atualmente têm se aproveitado para otimizar os seus resultados em resolução de entidades.

O ser humano tem facilidade para interpretar diversas representações de dados. Entretanto, o ideal é que as tarefas sejam simples, permitindo que qualquer pessoa possa realizar a tarefa (ZHANG *et al.*, 2014).

A abordagem mais atual, a híbrida, está sendo aplicada em muitos trabalhos (WHANG *et al.*, 2013, DEMARTINI *et al.*, 2012, GOKHALE *et al.*, 2014, ZHANG *et al.*, 2014, WANG *et al.*, 2012), de maneira que aproveita os pontos positivos da abordagem humana e computacional.

O trabalho descrito em GOKHALE *et al.* (2014) foi o primeiro a propor a ideia de *Hands-off Crowdsourcing* (HOC), que é uma ferramenta que não depende de um desenvolvedor. Esta abordagem supera o principal obstáculo das propostas híbridas, entretanto, ainda é muito limitada no quesito dos dados de entrada que podem ser processados pelo modelo.

3.8.2.Aplicações

A partir do ano 2012, diversos trabalhos na área de resolução de entidades foram publicados com o foco em *crowdsourcing*, buscando uma abordagem híbrida, onde a multidão contribui em algumas etapas do *workflow* da resolução, e o restante é executado por algoritmos provenientes da própria área de resolução de entidades. As aplicações a seguir são os resultados da revisão sistemática que se encontra no Apêndice 1.

3.8.2.1.ZenCrowd

ZenCrowd foi um sistema desenvolvido em 2012 para fazer uma combinação de técnicas de *crowdsourcing* e *reasoning* probabilístico na resolução de entidades, mais especificamente no ERA 3, em larga escala.

O sistema recebe como entrada um conjunto de páginas HTML que são passadas a um Extrator de Entidades, que identifica potenciais entidades textuais relevantes mencionadas na página. Uma vez detectadas, essas entidades são enviadas a um comparador algorítmico que tenta ligar automaticamente entidades textuais e outras

entidades semanticamente similares da LOD (*Linked Open Data*) *Cloud*. Pelo fato do carregamento ser muito demorado, foi criado um índice local para guardar informações relevantes da LOD *Cloud*. Os Combinadores Algorítmicos retornam listas com os *k* primeiros links das entidades da LOD.

Os resultados dos Combinadores Algorítmicos são armazenados em uma Rede Probabilística e são combinados e analisados utilizando técnicas de inferência probabilística. Os resultados são tratados de três modos diferentes, dependendo da sua qualidade. Caso o resultado do algoritmo de decisão seja muito alto, os resultados são automaticamente armazenados em um banco de dados local. Caso os resultados sejam muito baixos, eles são descartados. E caso eles sejam considerados incertos (se os valores de confiança forem relativamente baixos, por exemplo), eles são passados para o Gerador de Micro Tarefas, que extrai fragmentos relevantes da página HTML original e gera dinamicamente micro tarefas usando um modelo pré-definido. As tarefas são publicadas na plataforma *crowdsourcing* para serem manuseadas pelos trabalhadores.

Os trabalhadores são classificados em *confiáveis* e *não confiáveis* a partir de uma rodada onde os resultados já são conhecidos previamente. A confiabilidade de um trabalhador é calculada dividindo-se o número de resultados corretos pelo número de resultados mostrados ao trabalhador, ou seja, *resultados_corretos / numero_resultados*. A confiabilidade é atualizada iterativamente à medida que o trabalhador avança no jogo.

Os dados para avaliação consistem de 25 notícias escritas em inglês vindas de diversas fontes. As notícias foram selecionadas de modo que houvesse assuntos de interesse global, e de diversos países, como Estados Unidos, Índia e Suíça. Foram extraídas 489 entidades utilizando o *Stanford Parser* (KLEIN & MANNING, 2003). Para os experimentos foram empregados 80 trabalhadores distintos da plataforma Amazon MTurk. Para cada tarefa era pago $0.01 que consistia em selecionar uma URL dentre cinco que houvesse relação com a entidade em questão.

Para avaliar a efetividade do método, foram computados dados de *precision* (P), *recall* (R) e *accuracy* (A) que foram calculados em função de verdadeiro-positivos, verdadeiro-negativos, falso-negativos e falso-positivos (Seção 3.7.4).

Na análise dos resultados, notou-se que a utilização das técnicas de *crowdsourcing* aumentou a precisão dos resultados em 6%. As vantagens do sistema são a existência do *framework* probabilístico que promove um aumento de desempenho

considerável, variando entre 4% e 35% em relação ao modo manualmente otimizado, e em cerca de 14% da otimização automática correspondente.

3.8.2.2.CrowdER

CrowdER (WANG *et al.*, 2012) é um sistema que se utiliza do poder da multidão para aprimorar a acurácia de métodos de Resolução de Entidades utilizando o trabalho humano.

Para comparar *strings*, o autor utilizou a distância de Jaccard, e para comparar valores numéricos, ele utilizou um certo valor de variação para mais ou para menos.

Pares que tinham similaridade abaixo de um certo *threshold* eram automaticamente descartados e os outros eram enviados aos "trabalhadores" na forma de pares ou de clusters. Nas tarefas em pares, eram apresentadas duas referências e era perguntado ao trabalhador se aquelas referências eram equivalentes ou não. Já no caso de *clusters*, era exibida um certo número de elementos e o trabalhador tinha que classificar os registros de acordos com as *labels* que eram atribuídas. Elementos com o mesmo *label* pertenciam ao mesmo *cluster*.

Nos experimentos foram utilizadas duas bases de dados: uma de produtos e outra de restaurantes. Os resultados mostraram altos valores de *recall* para *thresholds* muito baixos.

3.8.2.3.CrowdMatcher

CrowdMatcher é um sistema híbrido (máquina-multidão) que tem como objetivo o *schema matching*, que é a comparação de *schemas* de diversas bases de dados em busca de atributos que possam ser associados (ZHANG et al., 2014).

Os *matchings* gerados pela ferramenta são verificados através de questões disponíveis para a multidão (*Correspondence correctness queries* - CCQs), que determinam se correspondem ou não à mesma entidade.

Dado um par de esquemas, a primeira etapa é a utilização da ferramenta *schema matching* para gerar um conjunto de possíveis pares, cada um associado a uma probabilidade. Devido à incerteza, a probabilidade de possíveis pares é relativamente baixa (geralmente menos de 50%), e, para isso, o sistema reduz a incerteza colocando CCQs aos trabalhadores da multidão.

CrowdMatcher se utiliza da mesma estratégia de CrowdER (WANG *et al.*, 2012), adotando duas abordagens, CCQ únicos por HIT, e múltiplos CCQs, a fim de maximizar a redução de incerteza dentro de um orçamento limitado.

CrowdMatcher também inclui várias características, como: integração de diferentes *matchings* gerados a partir de ferramentas clássicas de *schema matching*; minimização dos custos de *crowdsourcing* selecionando automaticamente o conjunto mais informativo de CCQs dos possíveis *matchings*; é capaz de gerenciar respostas imprecisas fornecidas pelos trabalhadores; e as respostas da multidão são usadas para melhorar os resultados correspondentes.

Para o controle de qualidade, assumindo que nem todos os trabalhadores respondem corretamente a questão, a ferramenta lida com as taxas de erros utilizando o Teorema de Bayes (ZHANG *et al.*, 2013).

3.8.2.4.Corleone

GOKHALE *et al.* (2014) propõem um conceito chamado *hands-off crowdsourcing* (HOC). O trabalho de Gokhale concluiu que HOC consegue ser escalável para necessidades de *entity matching* para empresas, *startups* e sistemas *crowdsourcing*.

Corleone é uma solução que utiliza o conceito HOC para solucionar EM, e foi demonstrado no artigo de GOKHALE *et al.* (2014) que esta solução produziu resultados comparáveis ou até melhores, em relação às soluções tradicionais.

Ele se utiliza de regras para fazer um pré *matching* de pares com um certo nível de semelhança e gerar pares candidatos para serem enviados aos trabalhadores do Amazon MTurk.

As tarefas foram projetadas em forma de perguntas onde eram exibidos dois elementos e os trabalhadores deveriam analisar se os elementos representavam a mesma entidade ou não. Os *matches* são avaliados e os pares mais difíceis são repassados novamente aos trabalhadores para serem reavaliados.

Nos testes os autores utilizaram 3 bases de dados: restaurantes, citações e produtos. Para os testes, foram selecionados trabalhadores que possuíam altos valores de confiabilidade no Amazon MTurk. Os resultados obtidos tiveram alto grau de precisão e acurácia, com valores acima de 80%, em sua maioria.

3.8.2.5.Comparativo

De acordo com Tabela 8, é apresentado um comparativo entre os modelos e ferramentas retornados pela revisão sistemática. A etapa ERA 3 foi predominante nos casos estudados, mostrando ainda que embora a resolução de entidades seja um conjunto de atividades, o ERA3 ainda é a mais estudada nesse campo de pesquisa. Cada estudo concentrou seus esforços em uma abordagem diferente para o tratamento desta atividade.

A ferramenta CrowdMatchER (ZHANG *et al.*, 2014) tem como objetivo a associação de diferentes atributos de bases diferentes, de forma que essas informações são, na realidade, subsídios para futuras etapas, tanto em processos de resolução de entidade, quanto em outras áreas. Ou seja, este modelo se restringe especificamente no início do ERA3. Além disso, o CrowdMatchER não tem qualquer controle na qualidade dos dados por parte dos usuários.

A ferramenta Corleone (GOKHALE *et al.*, 2014), por outro lado, foca no processo de *Entity Matching*, que é a comparação dos valores de um mesmo atributo entre bases de dados. Corleone não se utiliza de nenhuma estratégia própria de controle de qualidade de dados ao longo do processo. GOKHALE *et al.* (2014) adotaram o controle de qualidade fornecido pelo MTurk.

ZenCrowd (DEMARTINI *et al.*, 2012) tem uma abordagem que lida com LOD (*Linked Open Data*), e não foca totalmente nas etapas de ER, pois transfere grande parte para o *LOD Cloud*. Na realidade, ZenCrowd faz uma chamada ao *LOD Cloud*, e este por sua vez executa o ERA3. A ferramenta possui estratégias de qualidade de dados, como a criação de um indicador de qualidade ao decorrer da tarefa assim como uma rodada de qualificação.

O CrowdER (WANG *et al.*, 2012) tem aspectos muito parecidos do Corleone, porém o CrowdER incorpora um controle de qualidade básico utilizando um HIT para qualificação.

Tabela 8 - Características das ferramentas de resolução de entidades com *crowdsourcing* (elaborado pelo autor)

Ferramenta	Etapa de ER	HOC	Dados de Entrada	Estratégias de Qualidade	Quem pode testar	HIT	Motivação da Multidão
Corleone	ERA3	Sim	Duas bases e atributos variados	-	Usuários do MTurk	Múltipla escolha (Sim, Não, Em dúvida)	Monetização por tarefa realizada
CrowdER	ERA3	Não	Duas bases e atributos variados	Único HIT para qualificação	Usuários do MTurk	Múltiplas escolhas por par / categorização múltipla	Monetização por tarefa realizada
CrowdMatchER	ERA3 (schema)	Não	Diversos *schemas* e atributos variados	-	Usuários do MTurk	Múltipla Escolha (Sim ou Não)	Monetização por tarefa realizada
ZenCrowd	ERA3	Não	Website	- Qualificação - Atualização de qualidade	Usuários do MTurk	Múltipla Escolha	Monetização por tarefa correta

3.8.3.Experimentos

Na área de Resolução de Entidades não existe nenhum procedimento formal para a avaliação de um modelo, nada relacionado a dados de entradas e nem métricas utilizadas após a execução dos algoritmos e as tarefas relacionadas a multidão.

Através dos trabalhos retornados pela revisão sistemática descrita no Apêndice 1, foi possível identificar uma padronização quanto a metodologia dos experimentos. A ferramenta ZenCrowd como mencionado na seção 3.8.2.1, tem como entrada WebSites. Já a ferramenta CrowdMatcher tem como entrada *schemas* de base de dados. As duas ferramentas têm focos diferentes de resolução de entidades em relação ao trabalho proposto, e por isso estes trabalhos não serão levados em consideração quanto aos dados utilizados em seus experimentos.

A ferramenta CrowdER e Corleone tem abordagens muito parecidas com o presente trabalho, e por esse motivo serão analisados a forma como conduziram os seus experimentos e também os dados utilizados. Além disso, o Desafio de Resolução de Entidades proposto por TALBURT *et al.* (2009) também será analisado neste estudo.

3.8.3.1.Experimento da ferramenta CrowdER

Dados de Entrada

O estudo de WANG *et al.* (2012) analisa duas bases de dados reais, a primeira relacionada à restaurantes e a segunda a produtos. Cada base contém dois arquivos, de origens distintas.

A Tabela 9 apresenta informações sobre as referências dessas duas bases. A base referente à entidade Restaurante contém os arquivos Fodors e Zagats, com 533 e 331 registros, respectivamente. Esta base apresenta 106 agrupamentos de restaurantes, indicando assim, 106 restaurantes únicos entre os dois arquivos.

Para a base referente à entidade Produto, os websites Abt e Buy contêm 1081 e 1092 registros, respectivamente. Esta base apresenta 1097 agrupamentos e, consequentemente, 1097 produtos únicos.

Tabela 9 – Perfil dos dados do estudo da ferramenta CrowdER (adaptado de WANG *et al.* (2012))

	Restaurante[35]		**Produto**[36]	
	Fodors	Zagats	Abt *Website*	Buy *Website*
Quantidade de Registros	533	331	1081	1092
Quantidade de agrupamentos	106		1097	
Atributos	[*name*, *address*, *city*, *type*]		[*name*, *price*]	
Exemplo de registro	[“oceana”, “55 e. 54th st.”, “new york”, “seafood”]		[“Apple 8GB Black 2nd Generation iPod Touch - MB528LLA”, “$229.00”]	

Os critérios de *matching* definidos por WANG *et al.* (2012) são bastante simples e são descritos abaixo:

1) Não existem referências equivalentes dentro de um mesmo arquivo;
2) Só existe no máximo um par de referências equivalentes entre os arquivos, ou seja, o tamanho máximo de um cluster é de duas referências e o tamanho mínimo, de um apenas.

Metodologia

CrowdER utiliza a métrica de *recall* para calcular a eficiência do seu modelo. Como o objetivo do estudo era a otimização do número de HITs, o estudo comparou o *recall* (Tabela 10) dos resultados em função da variação do número de pares que eram gerados devido a um *threshold* especificado.

O estudo provou que mesmo diminuindo o número de pares de 367.653 para 161, o modelo proposto ainda apresentava um *recall* de 78,3%.

[35] http://www.cs.utexas.edu/users/ml/riddle/data/restaurant.tar.gz

[36] http://dbs.uni-leipzig.de/file/Abt-Buy.zip

Tabela 10 – Exemplo de resultado de avaliação de times (adaptado de TALBURT *et al.* (2009))

Threshold	Total #Pair	Matches	Recall
0.5	161	83	78,3%
0.4	755	99	93,4%
0.3	4.788	105	99,1%
0.2	23.994	106	100%
0.1	83.117	106	100%
0	367.653	106	100%

3.8.4.Experimento da ferramenta Corleone

Dados de Entrada

O experimento de GOKHALE *et al.* (2014) consistiu de três entidades: Restaurante, Citação e Produto. As bases relacionadas à entidade Restaurante são os mesmo do estudo de WANG *et al.* (2012) já apresentado anteriormente.

A entidade Citação nesse estudo é composta pelas bases DBLP[37] e Google Scholar[38], que foram extraídas do estudo de KÖPCKE & RAHM (2010). Essa entidade, inclusive, é bastante usada em trabalhos de *Entity Matching* (GOKHALE *et al.*, 2014).

A base de dados da entidade Produto foi criada pelos próprios autores do estudo que extraíram os dados da Amazon e Walmart, sendo o objetivo dos autores selecionar um conjunto de dados diversificados aumentando a dificuldade de *matching*.

A Tabela 11 apresenta a quantidade de registros em cada tabela relacionada a cada base de dados.

[37] http://dblp.uni-trier.de

[38] http:// scholar.google.com

Tabela 11 – Base de dados para o experimento da ferramenta Corleone (adaptado de GOKHALE *et al.* (2014))

Base de Dados	Tabela A	Tabela B	Número de *Matches*
Restaurantes	533	331	112
Citações	2.616	64.263	5.347
Produtos	2.554	22.074	1.154

Metodologia

GOKHALE *et al.* (2014) analisaram o seu experimento segundo as métricas de *precision*, *recall* e F1-Score. A Tabela 12 apresenta um exemplo da aplicação da ferramenta Corleone em relação às bases de restaurantes, citações e produtos.

Nota-se também que, similarmente ao estudo da ferramenta CrowdER, os princípios de *matching* são os mesmos descritos na seção 0.

Tabela 12 – Desempenho da ferramenta Corleone (adaptado de GOKHALE *et al.* (2014))

Base de Dados	*Precision*	*Recall*	F1
Restaurantes	97%	96,1%	96,5%
Citações	89,9%	94,3%	92,1%
Produtos	91,5%	87,4%	89,3%

3.8.5.Experimento do Desafio de ER

Dados de Entrada

ZHOU & TALBURT (2011) desenvolveram um desafio que simula a realidade encontrada em projetos de integração de dados. Dos desafios propostos, alguns são: lidar com a falta de dados ou dados incompletos, avaliar e melhorar a qualidade dos dados, desenvolver uma estratégia de integração, selecionar e utilizar ferramentas adequadas, ensaios repetidos e trabalho em equipe.

Os dados utilizados neste desafio foram gerados pela ferramenta SOG (*Synthetic Occupancy Generator*) (TALBURT *et al.*, 2009) e são referências que representam clientes que residem nos Estados Unidos. As suas características são descritas abaixo (ZHOU *et al.*, 2013):

1) Lista A – Arquivo CSV contendo 94.306 registros. Os atributos dessa base são: ID, nome, rua do endereço, cidade-estado-zip, *PO Box*, *PO Box city-state-zip*, *social security number* (SSN) e data de aniversário;
2) Lista B – Arquivo delimitado por caractere contendo 100.777 registros. Os atributos são: ID, primeiro nome, último nome, número da rua, endereço primário, endereço secundário, cidade, *estado*, *zip code*, e número de telefone;
3) Lista C – Arquivo de campo de tamanho fixo contendo 76.059 registros. Os atributos são: primeiro nome, nome do meio, último nome, *social security number*, data de aniversário e número de telefone;
4) COA – Arquivo limitador por caractere contendo 40.174 registros. Os atributos são: último nome, endereço de origem (número, nome da rua, etc.) seguido do endereço de destino.

O quarto arquivo armazena a mudança de endereço dos clientes. Este arquivo interliga logicamente as três listas indicando o local de origem e destino da mudança de endereço de um cliente. O objetivo do desafio é criar grupos de referências equivalentes entre as três listas.

Metodologia

A métrica principal do Desafio de ER é o T-W Index (seção 3.7.1), entretanto outras métricas são utilizadas para avaliar os diferentes times que resolvem o problema. Uma delas é o *Group count* que representa o número de agrupamentos das referências. *Overlap count* representa a quantidade de interceções das referências de um resultado comparado ao gabarito. Por fim, *Average Group Size*, que é o tamanho médio dos grupos formados.

A Tabela 13 mostra o exemplo de uma coleção de resultados de cinco times que realizaram o Desafio de Resolução de Entidades.

Tabela 13 – Exemplo de resultados de avaliação de times no Desafio de ER (adaptado de TALBURT *et al.* (2009))

	True	Team 1	Team 2	Team 3
T-W Index	1.0000	0.6250	0.5227	0.4483
Group Count	20,067	41,418	65,185	17,541
Overlap Count	20,067	46,126	69,192	41,848
Avg. Group Size	13.5	6.5	4.2	15.4
Class Distribution				
1	200	12,180	28,500	200
2	400	8,078	21,700	400
3	2,300	1,300	4,000	2,000
4	3,800	2,500	3,250	3,600
5	4,500	3,560	2,500	4,500
6	5,617	7,000	2,485	3,800
7	2,450	6,000	2,450	2,141
8+	800	800	300	900

É possível verificar que o Time 1 obteve o melhor desempenho dentre todos os times, pois o grau de similaridade de seu resultado chegou a 62,5%.

Outros indicadores utilizados por TALBURT em seu desafio, é a distribuição de classes (*class distribution*), indicada na Tabela 13. A distribuição de classes tem como objetivo calcular a quantidade de registros por tamanho de cluster. Este indicador apresenta como estão sendo distribuídos os registros em relação ao seu tamanho. Por exemplo, na tabela acima, os resultados do Time 3 apresentam 4500 referências equivalentes com três registros.

3.9.Considerações Gerais

Através dos estudos deste capítulo, é possível afirmar que a área de Resolução de Entidades é extremamente ampla, e com o passar dos anos adquiriu e incorporou diversos outros conceitos dentro de seu processo. Também é inevitável dizer que a Resolução de Entidades tem importância fundamental no processo de melhoria na qualidade de dados, seja em momentos de Integração de Dados, análise utilizando mineração de dados e *business intelligence*.

Assim como a maioria dos processos, a resolução de entidades utilizava essencialmente algoritmos computacionais, tanto para comparar os atributos entre as entidades, quanto no aprendizado, aplicando regras a fim de determinar referências equivalentes. Entretanto, os sistemas *crowdsourcing* estão cada vez mais presentes também na área de ER. Neste capítulo foram descritos diversos estudos e trabalhos que propõem uma abordagem híbrida, utilizando tanto a multidão quanto algoritmos computacionais.

Entretanto, é possível observar que nas abordagens propostas a multidão tem sua capacidade bastante limitada, de modo que somente são criadas tarefas de complexidade extremamente baixa. Este trabalho propõe uma abordagem em que a multidão se torne mais presente no processo de resolução de entidades. Esta atitude possibilita tornar o modelo mais genérico de forma a abranger mais situações, como detectar referências equivalentes a partir de referências que não necessariamente têm exatamente os mesmos atributos. Tal liberdade ainda permite a multidão avaliar referências que contenham outros tipos de mídias, como áudio e vídeo, por exemplo. Entretanto, esta liberdade em um primeiro momento, contribui para a diminuição da qualidade dos dados, porém o modelo proposto foi projetado para atentar a essa questão tão importante.

4.Proposta

A resolução de entidades é um processo extremamente importante em áreas como integração de dados, mineração de dados, *business intelligence*, entre outros. Embora o objetivo principal da resolução de entidades seja a detecção de referências equivalentes, tais ações contribuem para a eliminação de dados duplicados, e até mesmo inconsistências em etapas futuras. Com essa capacidade, a resolução de entidades prepara os dados fazendo com que as análises posteriores sejam mais fiéis à realidade, aumentado as chances de sucesso.

A multidão é uma mão-de-obra diferenciada, criativa e, acima de tudo, adaptável às diversas situações, com um grande potencial evolutivo, conforme relatado na Seção 2.1. Muitas ferramentas estão se utilizando desse potencial para aumentar a acurácia na resolução de entidades. A principal delas é a abordagem híbrida, onde parte do processo é realizada pela multidão e parte realizada por algoritmos, geralmente de aprendizado, como mostrado na Seção 3.8.2.

A proposta deste trabalho é propor um modelo que seja viável, no sentindo de criação de uma ferramenta prática e também viável em termos de eficiência. O modelo que será apresentado, também contempla uma maior participação da multidão no processo de resolução, fato este, que não ocorre em outras abordagens estudadas no capítulo anterior. A participação da multidão possibilita o modelo a tratar muito mais casos de referências do que as abordagens estudadas na revisão sistemática.

4.1.Motivação

Através dos resultados da revisão sistemática, foi possível avaliar a tendência dos trabalhos nesses últimos anos. Uma síntese das abordagens dos quatro estudos selecionados para comparação é que todos eles utilizam a multidão, mas não em sua plenitude, como pode ser lido na Seção 3.8.2.5.

As abordagens mais recentes de resolução de entidades com *crowdsourcing* limitam a multidão em resolver somente partes bem específicas do workflow de ER, e as partes restantes, que normalmente são algoritmos de aprendizados, ficam sob responsabilidade de um desenvolvedor (GOKHALE *et al.*, 2014).

O trabalho de GOKHALE *et al.* (2014) é o que mais se aproxima de uma abordagem que aproveita melhor a capacidade da multidão, porém o contexto de

resolução de entidades deste trabalho também se encontra com muitas restrições, não sendo possível fornecer mais de duas bases simultaneamente.

Todo este processo não é tão simples quanto parece. A multidão pode sim produzir resultados satisfatórios, mas é importante atentar à qualidade dos dados produzidos. Esse é um aspecto crítico para a resolução de entidades, uma vez que ER está intrinsicamente ligado à qualidade dos dados.

O objetivo principal deste trabalho é criar um modelo que fomenta a participação da multidão de forma maximizada no processo de resolução de entidades. Um exemplo simples, é que os dados de entrada para os modelos e ferramentas estudados contemplam no máximo duas listas onde são comparadas uma a outra em busca de referências equivalentes. O modelo proposto por este trabalho busca agregar os mais diversos tipos de entradas de referências para a resolução de entidades, sem a perda de qualidade dos resultados gerados.

Além disso, o modelo inclui estratégias para garantir a qualidade dos dados, condição essencial para uma resolução de entidade eficiente. O modelo tem como foco a atividade ERA3 explicada na introdução da Seção 3.4.3, considerada a atividade principal de ER.

4.2.Definições

Dois papéis estão envolvidos no modelo. O primeiro é o responsável pela aplicação, é aquele que tem o interesse em agrupar as referências de modo a interligar as referências equivalentes, mais especificamente aplicar a atividade ERA3 do processo de TALBURT (2010) com o máximo de qualidade e menor custo.

O segundo papel são os usuários provenientes da multidão, i. e., são as pessoas que participarão das atividades para resolver a resolução de entidades propriamente dita.

Durante a tarefa, serão manipulados basicamente dois tipos de referências, ambos extraídas das bases de dados selecionadas pelo responsável. O primeiro tipo de referência é a chamada de alvo, e são referências que a multidão terá como parâmetro para buscar as referências equivalentes. A outra é a referência suspeita, e são aquelas que serão agrupadas juntamente com a referência alvo, de modo que todas elas referenciam a mesma entidade.

4.3.Etapas

O modelo contempla duas fases: (1) a fase de definição e (2) a fase de execução. A primeira é realizada pelos responsáveis, enquanto a segunda é realizada pelos supervisores e a multidão. A Figura 20 ilustra essas etapas, fases, com seus passos respectivos. Os detalhes de cada passo serão explicados logo a seguir.

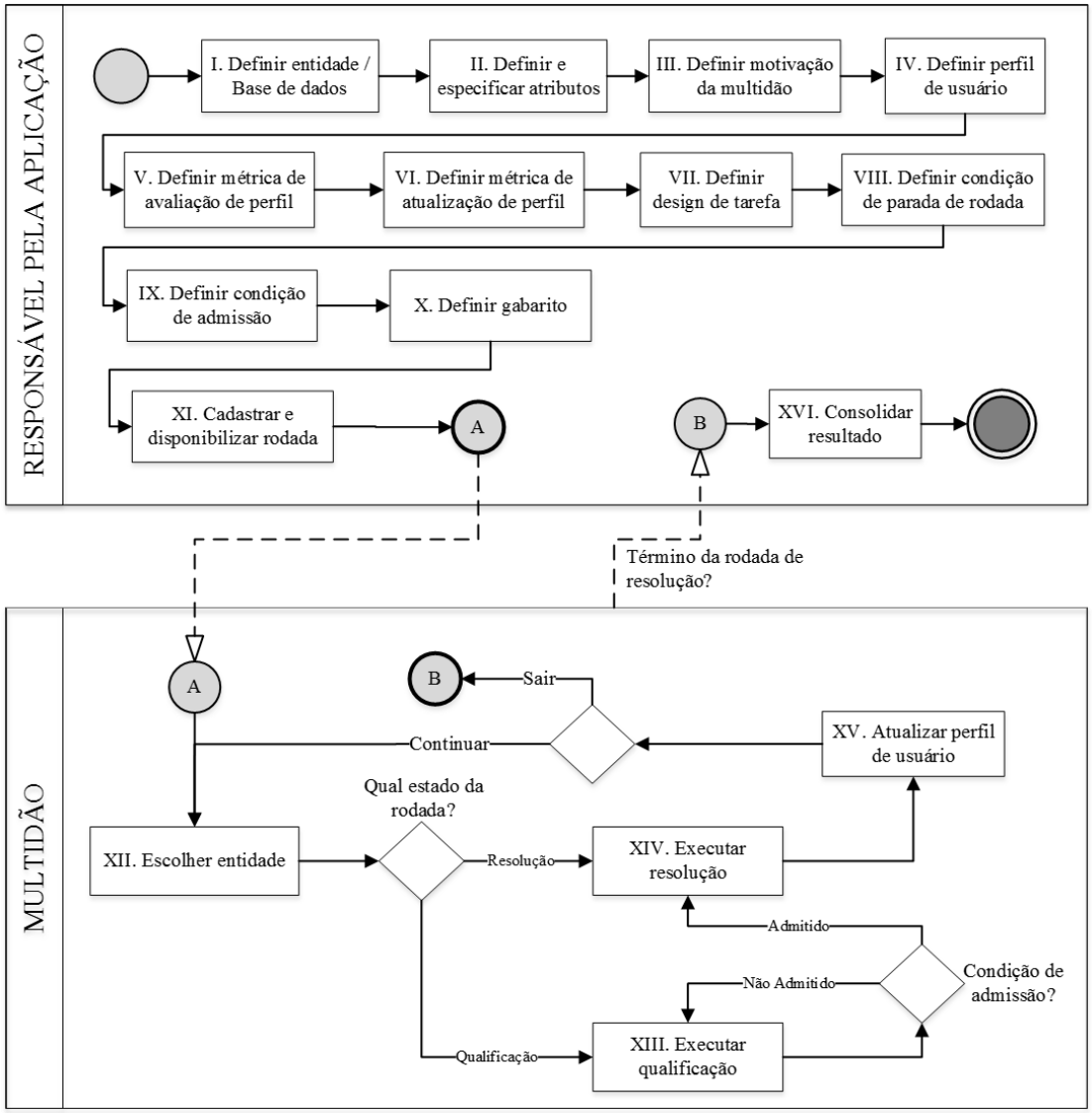


Figura 20 – Etapas e passos do modelo (elaborado pelo autor)

As etapas definidas no processo, na fase de definição, são bastante flexíveis, deixando a maioria das definições a cargo de quem for implantá-lo. Entretanto, existem alguns pontos rígidos que deverão ser respeitados:

- Visibilidade – A visibilidade das ações de um usuário, bem como sua resolução para uma determinada referência não são visíveis no sistema para outros usuários.
- Qualidade de Dados por Entidade – O usuário terá sua qualidade calculada para cada entidade a qual está associado. Caso tenha três entidades associadas, serão três estimativas de qualidade.
- Rodada – A rodada significa que a entidade estará disponibilizada para a multidão até a condição de parada ser satisfeita. A rodada tem duas fases: qualificação e resolução. A fase de qualificação antecede a de resolução e tem como objetivo avaliar o desempenho dos usuários.
- *Design* da tarefa – A tarefa consiste na escolha de uma referência alvo, que é a base para a busca de referências equivalentes. O objetivo é permitir ao usuário flexibilidade necessária para que a capacidade da multidão seja maximizada.
- Ferramenta de Busca de Referências – É essencial fornecer uma ferramenta que possibilite o usuário realizar as buscas das referências por conta própria. Seguindo a mesma linha de raciocínio do *design* de tarefa, o objetivo é flexibilizar as ações do usuário dando-lhe liberdade.
- Regras de *Matching* – As regras de *matching* devem ser exclusivamente criadas pelo próprio usuário, bem como *schema matching*. Entretanto, o responsável pode orientar o usuário segundo as suas necessidades.

As etapas do responsável são executadas apenas uma vez (Etapa I a Etapa XI e Etapa XVI). Entretanto, as etapas da multidão (Etapa XII, Etapa XIII, Etapa XIV e Etapa XV), por serem atribuídas a muitas pessoas utilizando em paralelo, não são executadas apenas uma vez, havendo concorrência entre as execuções de diferentes usuários.

4.3.1.Etapas

I. Definir entidade / Base de dados

O Responsável da aplicação deverá definir a entidade a ser trabalhada para a resolução de entidades. Após isso, deverão ser reunidos as bases de dados e arquivos que contenham as referências e, posteriormente, definir aqueles que participarão do processo de ER.

Muitas vezes uma empresa tem alguma priorização de entidades em relação a outras, assim como bases que são mais importantes ou emergenciais.

II. Definir e especificar atributos

Este passo é importante pois a multidão não conhece a entidade de antemão e nem seus atributos. O atributo pode ser de conhecimento público, como o CPF, porém, caso seja pouco conhecido e bastante utilizado pelo Responsável, é importante que a multidão entenda como esse atributo se relaciona à entidade.

Nesse momento é importante definir também os atributos de cada base de dados que serão exibidos à multidão. Essa decisão deve ser bem pensada, porque o julgamento da equivalência de uma referência a outra depende essencialmente dos dados exibidos nas tarefas.

III. Definir motivação da multidão

Esta etapa especifica quais serão as formas de motivações oferecidas aos usuários em troca da colaboração nas tarefas. A recompensa pode ser oferecida somente aos primeiros colocados de um ranking, ou pode ser monetária (por tarefa realizada). As diversas opções em relação à motivação podem ser escolhidas da seção 2.2.1.4.

IV. Definir perfil de usuário

O perfil do trabalhador representa as suas ações, objetivos e pretensões. Como comentado na seção 2.2.2.1, a análise do perfil do trabalhador é fundamental para determinar a qualidade dos seus dados. Normalmente, cada perfil tem um padrão de atuação e através desses padrões é possível determinar se um certo trabalhador é diligente, competente ou até mesmo um *spammer*.

O modelo proposto e o conceito de referência alvo dão suporte a alguns tipos de análises como o cálculo do *recall* e *precision*; outros pontos de vista também são possíveis como os verdadeiro-positivos, falso-positivos e falso-negativos. Os resultados verdadeiros-negativos também podem ser calculados, entretanto, para o processo de ER, esse dado não é interessante.

Além desses indicadores, a acurácia, a eficácia, o tempo de resposta nas tarefas e o cálculo de similaridade de Talburt-Wang Index também são essenciais para definir o padrão de atuação dos usuários.

V. Definir métrica de avaliação de perfil

O trabalhador é peça fundamental em um sistema *crowdsourcing*. A avaliação do perfil em termos de qualidade dos dados produzidos é algo que, se conseguido com sucesso e exatidão, pode ser uma poderosa ferramenta para o aumento da qualidade dos dados do sistema.

A métrica será definida para duas situações distintas. A primeira é relativa à fase de qualificação. Nesta etapa do processo o gabarito é conhecido, logo é possível uma métrica mais exata para o cálculo da qualidade. A segunda situação é para a rodada de resolução, onde os resultados não são totalmente conhecidos e estão disponíveis apenas os resultados de outros usuários para comparação.

A métrica é uma fórmula ou algoritmo que deverá se basear em alguns componentes da resolução de entidades, como: *recall* e *precision*. Entretanto, a simples relação dessas duas medidas não representa a complexidade dos processos de ER. As medidas mais adequadas para este caso seriam os verdadeiro-positivos, falso-positivos e falso-negativos. Além disso, na área de ER, falso-positivos são piores do que falso-negativos, como já citado no Princípio 4 da Seção 3.3. É preferível a criação de falso-negativos do que falso-positivos, isto é, na dúvida o melhor é não arriscar. Logo, o recomendável é que seja uma fórmula ou algoritmo que contenha esses elementos e tenha a possibilidade de ajustar pesos.

VI. Definir métrica de atualização de perfil

O usuário deverá ter sua qualidade calculada em diferentes momentos, e isso pode ser definido aleatoriamente, ou a cada número de tarefas executadas. A atualização do perfil é fundamental para detectar *spammers* semi-aleatórios, aqueles

que em etapas de qualificação o fazem diligentemente, mas em outras etapas aleatoriamente. Além disso, a manutenção do perfil de um usuário reflete nos resultados ao longo das rodadas, pois o perfil do usuário determina a relevância de seus resultados no cálculo do voto pela maioria.

VII. Definir design de tarefa

O design de tarefa especificado nesta etapa será o mesmo aplicado à fase de qualificação e de resolução. Esta etapa é a mais crucial para a multidão, pois o *design* determinará o nível de complexidade na manipulação das referências. A facilidade no entendimento da tarefa implica em a multidão produzir melhores resultados.

Neste caso, é importante aplicar elementos que permitam flexibilidade, liberdade e simplicidade para que a multidão consiga ter acesso aos dados. Os usuários terão a liberdade de escolher a regra que desejarem para julgar a equivalência de duas referências.

A parte rígida desta etapa é a apresentação de uma referência alvo e ferramentas que possibilitem a multidão encontrar as referências equivalentes.

VIII. Definir condição de parada de rodada

A rodada é formada por duas fases: a qualificação e a resolução. O início da rodada se dá quando esta é criada e publicada (Etapa XI) para a multidão ter acesso. É interessante que a rodada criada para uma entidade seja limitada de alguma forma.

A condição de parada da rodada pode ser uma quantidade mínima de referências alvos a serem resolvidas. A condição também pode não só relacionar referências alvos, mas qualquer referência que foi agrupada, seja alvo ou suspeita.

Além disso, a condição de parada pode ser temporal, pois o responsável pode ter cronograma a cumprir com uma data prazo de entrega, ou ainda pode contemplar as duas restrições.

IX. Definir condição de admissão

Ao realizar a qualificação, o sistema definirá o perfil do usuário referente a uma entidade específica. A condição de admissão pode ser criada através de um *threshold*,

ou seja, caso o usuário não alcance um nível de acurácia desejado em relação aos casos de resolução de entidades, este não poderá continuar na fase de resolução.

Além disso, o responsável pode definir um número limitado de tentativas proibindo o usuário de prosseguir para a próxima fase de forma vitalícia. Ou ainda, permitir que usuário tente quantas vezes desejar.

X. Definir gabarito

A definição do gabarito é essencial para a rodada de qualificação, pois através dos dados do gabarito será possível ter uma avaliação fiel do perfil dos usuários. O gabarito é uma pequena parcela das referências suspeitas que serão agrupadas previamente.

XI. Cadastrar e disponibilizar rodada

Até está etapa, o responsável já terá definido todas as estratégias e técnicas que serão utilizadas durante toda execução do modelo, restando apenas a criação e a disponibilização da rodada.

A estratégia definida para a condição de parada da rodada (Etapa VIII), bem como a entidade selecionada para o processo de ER (Etapa I) são associados a uma rodada juntamente com os usuários que participarão. A rodada é então disponibilizada para que os usuários possam efetuar a resolução de entidades e contribuir com dados.

A escolha dos usuários nessa etapa pode influenciar, em muito, a qualidade dos resultados produzidos. Após a criação de algumas rodadas, já é possível conhecer o perfil dos usuários. Logo, é interessante para o responsável adicionar usuários com perfil de qualidade superior para uma entidade que contenha referências mais complexas ou que necessite de um grau maior de atenção.

XII. Escolher entidade

Após todas as definições realizadas pelo responsável, a multidão tem o seu início no processo. Como o modelo suporta a iteração de diversas entidades ao mesmo tempo, o usuário deverá escolher uma para iniciar. Como será explicado posteriormente, a entidade só poderá ser executada primeiramente na fase de qualificação e depois na fase de resolução.

XIII. Executar qualificação

O primeiro contato da multidão com a entidade é através da qualificação. Esta precede a etapa principal de resolução de entidades. Além de filtrar usuários com baixo rendimento, esta etapa também tem o intuito de treinar a multidão. Uma vez que as regras de *matching* podem não ser bem entendidas pelos usuários, a execução da qualificação é uma oportunidade para a multidão aprender implicitamente.

XIV. Executar resolução

Após a qualificação do usuário, ele está apto para executar a tarefa de resolução. Nesta etapa será gerada uma referência alvo que será a base para as buscas de referências suspeitas. O objetivo principal é agrupar todas as referências suspeitas com a alvo utilizando a regra de *matching* recomendada pelo responsável, e claro, através também de seus próprios conhecimentos.

XV. Atualizar perfil de usuário

Cada vez que a resolução é executada, os resultados dos usuários são analisados. Esta medida visa aplicar as técnicas definidas na Etapa VI, atualizando o perfil do usuário. A manutenção do perfil é essencial, pois o usuário pode ter um desempenho baixo no início da rodada e ao longo do tempo melhorar suas habilidades. Outro caso é que o usuário não tenha entendido bem o objetivo da tarefa e conforme o andamento, ele adquira o conhecimento necessário, executando assim a tarefa de forma correta.

XVI. Consolidar resultado

Ao término da rodada segundo as condições de parada definidas na Etapa VII, os resultados produzidos pela multidão são analisados e consolidados, formando assim a resolução de entidades. A consolidação será feita mediante alguma técnica ou algoritmo que leve em conta o perfil de cada usuário. Aqueles com baixa qualidade terão pouca relevância em seus resultados, porém usuários competentes ou diligentes terão bastante participação na influência dos resultados finais.

4.4.Ferramenta

A fim de demonstrar que o modelo é funcional e realizável, foi criada uma plataforma de resolução de entidades pela multidão. A ferramenta possibilita a inclusão de diversos tipos de dados (relacionados a pessoas e produtos, por exemplo) para a resolução de entidades através da multidão. A ferramenta recebeu o nome CERM, acrônimo para *Crowdsourcing Entity Resolution Model*.

Esta ferramenta é inspirada em modelos propostos em (WANG et al., 2012) e (WHANG *et al.*, 2013) que realizam uma abordagem híbrida, utilizando tanto a capacidade computacional quanto a humana. Entretanto, o foco está na maior liberdade da multidão encontrar referências equivalentes com as metodologias estudadas na área de qualidade de dados dando suporte ao modelo de modo que os resultados produzidos sejam ótimos.

4.4.1.Tecnologias

A plataforma foi desenvolvida utilizando a linguagem PHP em sua versão 5, lançando mão do conceito de MVC (Mode-view-controller). Para a camada de controle e visão, foi utilizado o framework Bootstrap[39] 3.2.0. O banco de dados utilizado é o Microsoft SQL Server Express 2008 R2 e o servidor web foi o Internet Information Services 7.0 (IIS). A API do JQuery[40] 2.1.1 foi utilizada em diversas partes da ferramenta, principalmente no recurso *Drag and Drop* nas tarefas executadas pelos usuários. Para a exibição de alertas e caixas de diálogos foi utilizada a API Messi[41], para o bloqueio de tela foi usada a API blockUI[42], para a criação de tooltips que exibem detalhes de certo elemento na página foi utilizada a API Tipsy[43] e para criação do cronômetro regressivo utilizado na tarefa foi utilizada a API Time Circles[44].

O modelo de dados está presente nos Apêndices 2 e 3. O Diagrama de Classe e Caso de Uso da ferramenta CERM estão descritos nos Apêndice 4 e 5.

---

[39] http://getbootstrap.com

[40] http://query.com

[41] http://marcosesperon.es/apps/messi

[42] http://malsup.com/jquery/block/

[43] http://onehackoranother.com/projects/jquery/tipsy/

[44] http://plugins.jquery.com/timecircles/

Na Figura 21 é apresentada a tela inicial da ferramenta CERM.

Figura 21 – Tela inicial da ferramenta CERM (elaborado pelo autor)

O módulo administrativo foi criado para dar suporte ao responsável aos dados. Neste módulo é possível cadastrar os usuários, criar e disponibilizar rodadas, cadastrar entidades e importar as bases de dados.

Como será elucidado a seguir, o fluxo inicia com o cadastro da entidade e suas bases de dados (em formato CSV). Ao serem importados, os dados dos arquivos são copiados para a base de dados interna da ferramenta CERM (Apêndice 2) criando assim tabelas dinâmicas para o armazenamento dos diferentes tipos de dados.

Na Figura 22 apresenta o Painel Administrativo da ferramenta CERM, que é acessado pelo responsável da aplicação.

Figura 22 – Módulo Administrativo (elaborado pelo autor)

Similar ao painel administrativo, os usuários também possuem um painel onde estão reunidos todos os dados sobre as tarefas disponíveis, bem como um ranking para instigar a competição entre os usuários.

4.4.2.Etapas

I. Definir entidade / Base de dados

O planejamento propriamente dito da definição da entidade e de suas bases de dados não é contemplada na plataforma, pois a ferramenta não tem um Driver ODBC (*Open Database Connectivity*) para acessar e estabelecer conexão com os diferentes SGBD disponíveis no mercado. Então, o responsável terá que realizar esta etapa externamente. A incorporação de um driver era de certa forma simples de ser realizada, porém como o foco do trabalho foi a criação do modelo para ER, não foi direcionado esforço para tal. A ferramenta, porém, permite importar arquivos no formato CSV, contendo os dados de interesse do responsável. Após a definição das entidades e das bases que participarão da resolução de entidades, o responsável deverá cadastrá-las no sistema.

Após a fase de execução, o sistema não permite a inclusão de novas bases de dados relacionadas a entidade em questão. Essa restrição é fundamental, pois a inclusão de uma base durante a fase de execução comprometeria a qualidade dos dados, elevando o número de falsos negativos.

A ferramenta não possui limitação quanto à quantidade de bases cadastradas para uma entidade, nem quanto ao número de registros. A Figura 23 apresenta a tela de cadastro de entidades na ferramenta CERM.

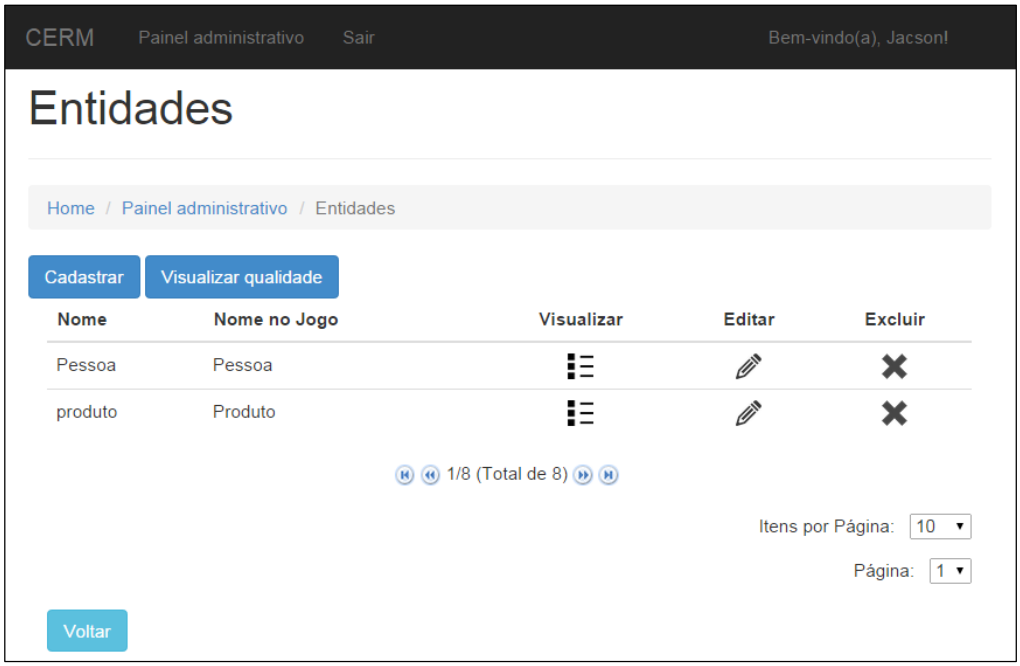


Figura 23 – Módulo Administrativo – Entidades (elaborado pelo autor)

II. Definir e especificar atributos

Ao cadastrar a entidade e posteriormente relacionar as bases de dados, a ferramenta permite a seleção dos atributos que farão parte do *design* da tarefa, isto é, os atributos que a multidão terá acesso e, portanto, tomará como base para definir as referências equivalentes (Figura 24).

Nessa etapa também a ferramenta permite a inclusão de um nome mais amigável e descrição para os atributos. Essas duas medidas auxiliam a multidão a comparar duas referências com mais precisão. Além disso, como é possível a inclusão de múltiplas bases no sistema, a descrição associada aos atributos permitirá a multidão criar associações tornando a resolução de entidades mais eficaz entre as bases.

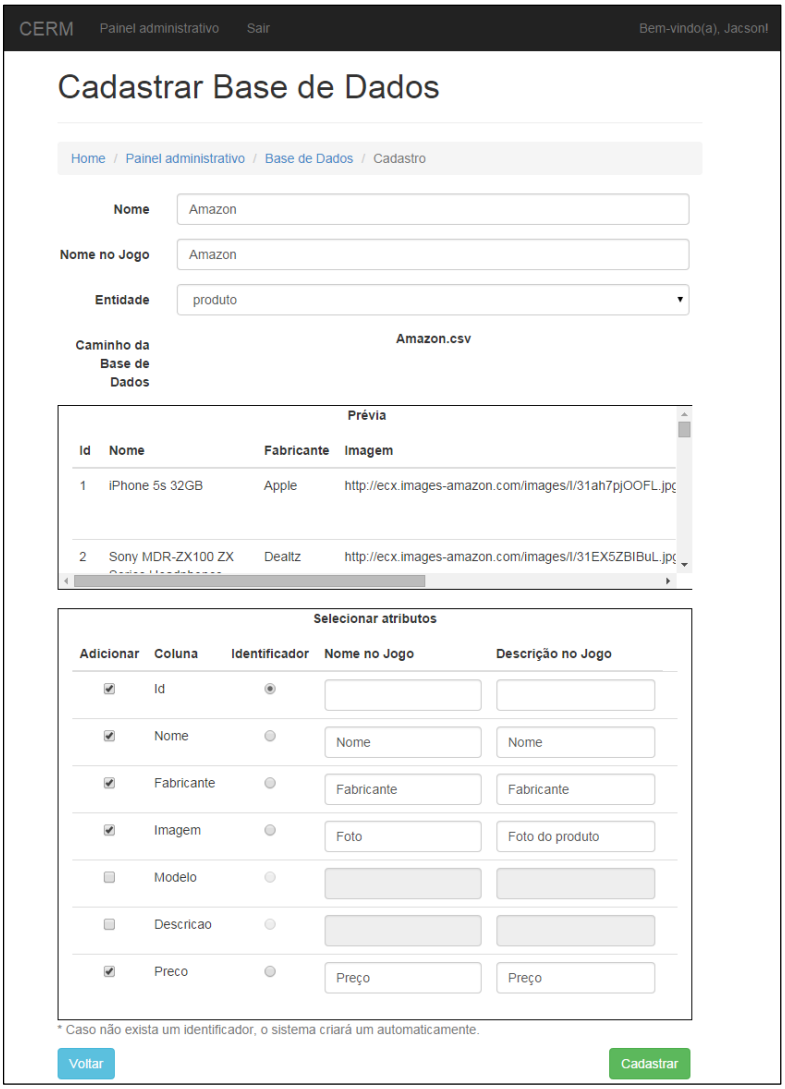


Figura 24 – Inserção de nome e descrição dos atributos da base de dados importada (elaborado pelo autor)

III. Definir motivação da multidão

A ferramenta depende essencialmente de três formas de motivação: recompensa, competição e altruísmo. Os usuários têm uma pontuação para cada entidade, e têm outra, global, que dá mais dinamismo à competição.

O ranking é sempre exibido no painel principal do sistema, logo após o usuário realizar a autenticação. A recompensa pode ser distribuída para os n primeiros colocados e o sistema pode informar a premiação.

A pontuação ocorre em dois momentos distintos, a primeira na fase de qualificação, onde a ferramenta CERM admite somente os usuários que conseguirem mais de 60 pontos.

O segundo momento é na fase de resolução, onde para cada tarefa executada é gerada uma pontuação. Entretanto, nesta fase, o gabarito ainda não é conhecido, logo uma estratégia para calcular a pontuação, diferente da fase anterior (qualificação), se faz necessária. A forma como será calculada essa pontuação é baseada no voto da maioria (RAYKAR *et al.*, 2010), onde a resolução resolvida será comparada a de outros usuários.

IV. Definir perfil de usuário

A ferramenta CERM adota a classificação descrita por KAZAI *et al.* (2011), onde os trabalhadores são definidos como: *spammer*, descuidado, incompetente, diligente e competente. Essas características serão definidas de acordo com o tempo de execução da resolução, *recall*, *precision* e tentativas de fraudes.

V. Definir métrica de avaliação de perfil

Para a rodada de qualificação onde o gabarito já é conhecido, a métrica utilizada pela ferramenta é o F-Score (seção 3.7.4) que é baseada nos resultados verdadeiro-positivos, falso-negativos e falso-positivos. O $F_{0,5}$ foi utilizado para calcular o agrupamento dos resultados dos usuários com o gabarito. O coeficiente *beta* com valor 0,5 significa que os resultados falso-positivos influenciam de maneira negativa no cálculo. Logo, quanto maior a quantidade de falso-positivos, menor será a qualidade dos resultados.

VI. Definir métrica de atualização de perfil

O cálculo de qualidade na rodada de resolução é mais delicado, pois o gabarito não está disponível. Logo, se faz necessário outra estratégia além da média F-Score. A estratégia empregada, nesse caso, foi a criação de tarefas com referências alvo cujo gabarito era previamente conhecido a cada 10 tarefas executadas por parte do usuário. Isto é, cada vez que um usuário realizar 10 resoluções, a próxima será um resultado conhecido, possibilitando um cálculo mais exato da qualidade da multidão.

VII. Definir design de tarefa

O design de tarefa da ferramenta CERM conta com o painel do usuário e a tarefa propriamente dita. O painel exibe um ranking com a pontuação global de todos os usuários, aumentando assim a competitividade e dando também acesso à tarefa de resolução.

A Figura 25 mostra um exemplo de uma tarefa na ferramenta CERM. A tarefa é formada por quatro componentes: (1) o filtro de busca – ferramenta onde são realizadas as buscas de acordo com os atributos. Retorna as referências suspeitas onde serão agrupadas a referência alvo. (2) grupo alvo – representado por uma área tracejada por uma linha azul, é o local onde serão agrupadas todas as referências equivalentes. (3) área de trabalho – é um local temporário onde o usuário poderá manipular as referências ao decorrer da tarefa. Quando a tarefa é finalizada, somente as referências dentro do "grupo alvo" serão associadas. (4) botões de ação – existem três botões na parte superior: o primeiro é de desistência; o segundo é de finalização; e o último de limpeza, isto é, a tarefa será reiniciada, voltando assim ao seu estado inicial.

O painel do usuário também contém diversas informações sobre as tarefas, que diz respeito a pontuações, a maneira como duas referências devem ser comparadas para serem equivalentes. A tarefa tem um tempo limite de 5 minutos para ser realizada.

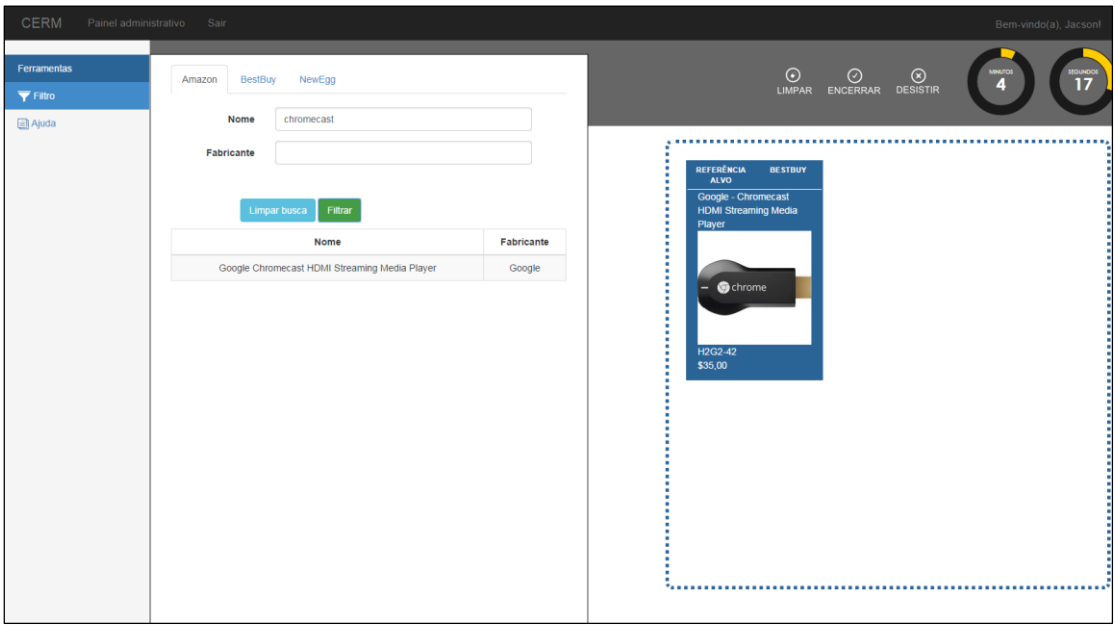


Figura 25 – Utilização da Ferramenta de Busca da Tarefa (elaborado pelo autor)

IX. Definir condição de admissão

A ferramenta CERM permite a aprovação do usuário somente se a pontuação gerada na qualificação ultrapassar 60 pontos. Caso não consiga esta pontuação, ele poderá tentar novamente quantas vezes desejar.

X. Definir gabarito

A ferramenta CERM conta com um módulo similar ao da Etapa I, onde é possível a importação de um arquivo CSV. Este módulo é próprio para a importação de gabarito da entidade. A única restrição é que os ids desse arquivo devem ser os mesmos *ids* importados na Etapa I.

XI. Cadastrar e disponibilizar rodada

A ferramenta permite o cadastro de rodadas e a associação de uma entidade a diversos usuários. O cadastro da rodada consiste na definição da data início de disponibilização e data prazo, quando será finalizado. Cada rodada tem apenas uma entidade associada. Uma vez iniciada, não é mais possível a inclusão de novas bases de dados para a entidade associada à rodada. O responsável também pode encerrar a rodada no momento que desejar, bastando acessar o painel administrativo.

XII. Escolher entidade

Esta é a primeira etapa em que os usuários participam efetivamente do sistema. A escolha da entidade ocorre através do painel de usuários. São exibidas somente as entidades habilitadas para o usuário.

O usuário pode executar as tarefas de duas entidades concorrentemente, isto é, enquanto em uma entidade o usuário está na fase de qualificação, em outra pode estar na fase de resolução. Todas essas informações podem ser acompanhadas pelo painel.

A Figura 26 mostra o painel principal onde o usuário visualiza todas as entidades cadastradas as quais ele pode ter acesso. Este acesso é definido pelo responsável da aplicação.

Figura 26 – Painel do usuário (elaborado pelo autor)

XIII. Executar qualificação

O primeiro contato da multidão com a entidade é através da qualificação. Esta precede a etapa principal de resolução de entidades. Além de filtrar usuários com baixo rendimento, esta etapa também tem o intuito de treinar a multidão. Uma vez que as regras de *matching* podem não ser bem entendidas pelos usuários, a execução da qualificação é uma oportunidade para a multidão aprender implicitamente com os seus erros. A pontuação gerada a cada tarefa apoia o aprendizado, uma vez que informa o desempenho ao usuário.

XIV. Executar resolução

Após o usuário ser qualificado e passar nas condições de admissão, ele está apto para executar a tarefa de resolução. Nesta etapa será gerada uma referência alvo que será a base para as buscas de referências equivalentes. O objetivo principal é agrupar todas as referências suspeitas com a alvo utilizando a regra de *matching* recomendada pelo responsável.

Ao final de cada tarefa, a ferramenta CERM executa o algoritmo descrito na seção **Erro! Fonte de referência não encontrada.** gerando assim um gabarito baseado os resultados da multidão. Este gabarito é comparado ao resultado do usuário através da média F-Score, gerando assim sua pontuação.

O usuário pode executar a tarefa quantas vezes desejar, e o sistema inclusive estimula essa prática. Quanto mais dados gerados, mais consolidados são os resultados.

XV. Atualizar perfil de usuário

A cada final de tarefa é aplicado o algoritmo de atualização de perfil. Na ferramenta ela ocorre de forma automática, sem a intervenção do responsável pela aplicação

Dependendo da estratégia definida, a atualização do perfil pode ocorrer a cada final de tarefa. O modo como ocorrerá essa atualização também depende da definição da etapa VI. Os perfis são atualizados de acordo com o seu rendimento, sendo analisados sob diversos aspectos. Essa etapa é executada automaticamente pelo sistema.

XVI. Consolidar resultado

Após o término da rodada, o responsável pelos dados acionará a opção de consolidação dos mesmos, disponibilizado pela ferramenta. Esta por sua vez executará o algoritmo de votação pela maioria descrita na seção **Erro! Fonte de referência não ncontrada.**. Porém, aplicando, todas as referências da entidade, e não somente uma específica, como ocorre na etapa XIV.

O algoritmo de consolidação leva em conta o perfil dos usuários, não somente o seu perfil mais recente. Entretanto, para os resultados realizados no passado, o algoritmo associa estes resultados ao perfil do usuário naquele momento. Assim, a relevância de um usuário competente, que no início das tarefas era descuidado, é maior na situação atual do que na passada.

Ao final, é disponibilizado um arquivo em formato CSV contendo as referências submetidas na etapa I com a inclusão de uma coluna *linking_id* que identifica os *clusters* formados.

Tabela 14 – Etapas do processo e técnicas correspondentes na ferramenta CERM (elaborado pelo autor)

Etapas do processo do Modelo	Técnica na ferramenta CERM
I. Definir entidade / Base de dados	Suporta parcialmente. Não é possível definir bases ou tabelas. Mas é possível importar arquivos CSV das bases.
II. Definir e especificar atributos	Ao importar o arquivo CSV, é possível selecionar os atributos que serão exibidos nas tarefas. Um nome amigável e a descrição de cada atributo podem ser informados pelo responsável.
III. Definir motivação da multidão	A motivação é a competitividade, recompensa e altruísmo.
IV. Definir perfil de usuário	*Spammer*, descuidado, incompetente, diligente e competente.
V. Definir métrica de avaliação de perfil	A métrica utilizada foi calcula pela média F1 Score.
VI. Definir métrica de atualização de perfil	A atualização do perfil é baseada no voto da maioria.
VII. Definir design de tarefa	O design possui componentes como ranking, pontuação, *drag and drop*, um filtro dinâmico de buscas e área de trabalho.
VIII. Definir condição de parada de rodada	A condição de parada da ferramenta CERM é temporal (data prazo) ou pela inativação realizada pelo responsável.
IX. Definir condição de admissão	O usuário é admitido se o seu desempenho ultrapassar 60 pontos. Abaixo disso, a tarefa de qualificação deve ser executada novamente.
X. Definir gabarito	A ferramenta CERM suporta a importação de arquivo CSV contendo gabarito.
XI. Cadastrar e disponibilizar rodada	O cadastro consiste na associação da entidade, dos usuários, de uma data de inicialização e término. A disponibilização da rodada ocorre no próprio sistema.
XII. Escolher entidade	Os usuários têm acesso ao painel do sistema onde estão localizadas todas as entidades disponíveis para as tarefas.
XIII. Executar qualificação	CERM também suporta a fase de qualificação, avaliando os usuários comparando seus resultados a um gabarito e gerando uma pontuação.
XIV. Executar resolução	Através do painel, o usuário executará as tarefas resolução que consiste no agrupamento de referências equivalentes em relação ao alvo.
XV. Atualizar perfil de usuário	O sistema atualiza o perfil definido na etapa VI de forma automática, após a execução de uma tarefa.
XVI. Consolidar resultado	A consolidação dos dados é realizada pelos dados gerados pelos usuários em função de seu perfil.

4.5.Considerações Gerais

O modelo proposto é composto por 16 etapas, sendo 11 executadas pelo responsável da aplicação e 5 etapas por parte da multidão. O desenvolvimento da ferramenta CERM mostra que o modelo proposto é viável.

A maior dificuldade encontrada no desenvolvimento do CERM foi planejar como o sistema seria construído de forma a atrair a atenção do usuário e tornar as tarefas menos complexas. A abordagem escolhida teve como foco tornar a tarefa (HIT) o mais intuitivo possível de modo a permitir o usuário a arrastar (*drag and drop*) os registros, tornar a busca mais prática e simples.

A utilização de um ranking dinâmico também foi proposta pela ferramenta CERM de modo a incentivar uma competição entre os usuários, com a finalidade de gerar mais resoluções em pouco tempo.

A Tabela 14 apresenta, de forma resumida, como as etapas do modelo foram aplicados particularmente a ferramenta CERM.

5.Experimento e Estudo de Caso

A partir da ferramenta criada, é possível comparar a eficiência do modelo aplicando as técnicas de cálculo de similaridade, como o T-W Index (TALBURT *et al.*, 2007), *recall*, *precision*, F-Score (GOUTTE & GAUSSIER, 2005), entre outros.

Neste capítulo serão apresentadas as metodologias que serão aplicadas para a realização do experimento, bem como as origens dos dados e o motivo para utilizá-las.

O objetivo do experimento é comprovar que o modelo proposto é viável e que a ferramenta CERM criada a partir do modelo apresenta eficiência muito próximas das ferramentas estudadas na revisão sistemática.

5.1.Metodologia

Através dos trabalhos retornados pela revisão sistemática descrita no Apêndice 1, foi possível identificar uma padronização quanto a metodologia dos experimentos. A ferramenta ZenCrowd como mencionado na seção 3.8.2.1, tem como entrada WebSites. Já a ferramenta CrowdMatcher tem como entrada *schemas* de base de dados. As duas ferramentas têm focos diferentes de resolução de entidades em relação ao trabalho proposto, e por isso estes trabalhos não serão levados em consideração quanto aos dados utilizados em seus experimentos.

Os trabalhos relacionados à resolução de entidades não têm à disposição uma base de dados formalizada para aplicação de modelos, algoritmos, *benchmarks* e afins. Logo, o estudo sobre os recentes experimentos se fez necessário para identificar as principais bases de dados utilizadas no contexto de multidão e resolução de entidades.

Os dados utilizados por WANG *et al.* (2012) são bastantes simples e, praticamente, não representam qualquer desafio para a multidão, ainda mais que as referências equivalentes se encontram uma em cada lista.

GOKHALE *et al.* (2014) criaram seus próprios dados relacionados à entidade Produto. Entretanto, os autores da ferramenta Corleone não disponibilizaram os dados de seus experimentos, tornando difícil uma comparação entre as ferramentas.

O desafio proposto por ZHOU & TALBURT (2011) seria uma coleção de dados excelente para o estudo de caso, entretanto, a base de dados do desafio não se encontra

disponível no Website do Centro para Resolução de Entidades e Qualidade da Informação da Universidade de Arkansas[45].

Uma forma de contornar este problema foi a tentativa de localizar a ferramenta SOG para simular os dados do desafio. Entretanto, a ferramenta também não se encontra disponível no website da universidade e em nenhum outro local como o SourceForge[46] ou GitHub[47].

Portanto, a solução para o estudo de caso foi a criação de uma base própria relacionada à entidade Produto. A respeito da métrica que será utilizada, será tomado como base as métricas do Desafio de ER (*T-W Index, Group Count, Avg. Group Size*), CrowdER (*recall*) e Corleone (*precision*, *recall* e *F-Score*).

5.2.Estudo de Caso

5.2.1.Motivação

A entidade Produto foi definida como alvo deste experimento, pois representa um tipo muito comum de entidade utilizada nos estudos relacionados à resolução de entidades. Como citado acima, tanto a ferramenta Corleone e CrowdER utilizam a entidade Produto em seus experimentos.

Outra justificativa é que os atributos relacionados aos produtos possuem bastante variação em seus dados, justamente pela diversidade desta entidade. Sendo ainda mais específico, foram selecionados apenas produtos eletrônicos para este estudo. O motivo é que, produtos eletrônicos são encontrados mais facilmente na Web e têm uma ampla divulgação de suas informações.

Com isso, foi criado um simples *Web Crawler* (PINKERTON, 1994) para a extração de informações relativas a produtos eletrônicos nos websites citados na seção a seguir.

---

[45] http://ualr.edu/eriq/

[46] http://sourceforge.net/

[47] https://github.com/

5.2.2.Fontes de Dados

Fontes de dados são bases de dados ou arquivos de onde as referências foram originadas. Para este experimento, foram escolhidas três fontes de dados: Amazon[48], BestBuy[49] e Newegg[50]. Estas empresas são três grandes redes dos Estados Unidos que comercializam principalmente produtos eletrônicos.

Estas três bases foram escolhidas pois possuem muitos produtos em comum, com fotos dos produtos de diferentes ângulos e descrições variadas. Notou-se também a duplicação de produtos dentro de um mesmo *website*.

A Tabela 15 apresenta os atributos associados a cada fonte de dados. Nota-se que o atributo "Nome" está presente em todas as três fontes. Entretanto, isso não é condição necessária para que o modelo possa funcionar, pois a multidão poderia fazer associação das referências através de outros atributos, inclusive pela imagem.

Tabela 15 – Atributos das três fontes de dados (elaborado pelo autor)

Atributos	Descrição	Amazon	BestBuy	NewEgg
Nome	Nome do produto	X	X	X
Modelo	Modelo do produto		X	X
Fabricante	Fabricante do produto	X		X
Descrição	Descrição do produto			X
Imagem	Imagem no formato JPG ou PNG referente ao produto	X	X	
Preço	Preço do produto em dólares		X	X

5.2.3.Entidade Produto

A Figura 27 é um exemplo de comparação entre duas referências de fontes diferentes, a esquerda originada da empresa Amazon e a direita da Bestbuy. A abordagem proposta no presente estudo permite que essas duas referências sejam ditas equivalentes.

[48] http://www.amazon.com/

[49] http://www.bestbuy.com/site/index.jsp

[50] http://www.newegg.com/

O ponto comum é a informação do modelo do notebook, “M6800”. Entretanto, na referência alvo esta informação está localizada no atributo “nome” do produto, enquanto essa informação está localizada no atributo “modelo” da referência suspeita.

As abordagens atuais e algoritmos computacionais precisariam ser configurados para incluir em suas regras essa possibilidade, comparando o atributo nome da fonte de dados Amazon com o atributo modelo da fonte de dados BestBuy.

Entretanto, essa é apenas uma das muitas possibilidades que um algoritmo deveria considerar para tratar todos os tipos de casos, o que torna inviável esse tipo de abordagem.

Figura 27 – Imagem extraída da ferramenta CERM, comparação de duas referências (elaborado pelo autor)

Além disso, a abordagem proposta, permite inclusive o uso de imagens, que são facilmente interpretadas pela multidão. Embora a imagem à esquerda esteja com pouca qualidade, a multidão ainda é capaz de associar as duas imagens como o mesmo produto. Por sua vez, um algoritmo computacional poderia concluir que os dois produtos são diferentes pelo fato do notebook à esquerda ter uma imagem de um automóvel em sua tela, e a da direita não.

Outra possibilidade é a inclusão de áudios e vídeos nas referências. Com a evolução tecnológica e facilidade do HTML-5, isso não seria problema nenhum para ser desenvolvido na ferramenta CERM.

Logo, quanto maior for o número de componentes agregados às referências, maior é a complexidade dos problemas que um algoritmo deve solucionar, enquanto para a multidão não passa de mais um simples campo.

5.3.Perfil dos dados

A seguir, a Tabela 16 apresentada o perfil dos dados extraídos dos websites definidos para o estudo de caso. O total de registros capturados para o experimento foram de 77 produtos. Se for considerar produtos únicos, isto é, descartando as referências equivalentes, o número é reduzido para 27 produtos. O tamanho médio de registros por produtos é de 2,85.

Tabela 16 – Perfil dos dados do experimento da ferramenta CERM (elaborado pelo autor)

	Amazon	**BestBuy**	**Newegg**	**Total**
Quantidade de registros	36	24	17	77
Quantidade de Grupos	27	22	13	27
Tamanho médio de grupos	1,33	1,09	1,31	2,85

5.4.Resultados

O experimento teve duração de cinco dias, iniciado em 15/09/2014 e com término em 20/09/2014. Nas seções seguintes os resultados obtidos serão descritos e representados em gráficos.

5.4.1.Participação de usuários

O número total de cadastros na ferramenta CERM foi de 42 pessoas. Dez pessoas apenas se cadastraram e não realizaram quaisquer atividades. Três tentaram passar na fase de qualificação, mas não tiveram êxito. Vinte e nove pessoas conseguiram chegar à fase de execução. A Figura 28 apresenta esses dados de forma ilustrativa.

Nota-se que a maioria dos usuários passou da fase de qualificação. A seguir esta fase será analisada com mais detalhes.

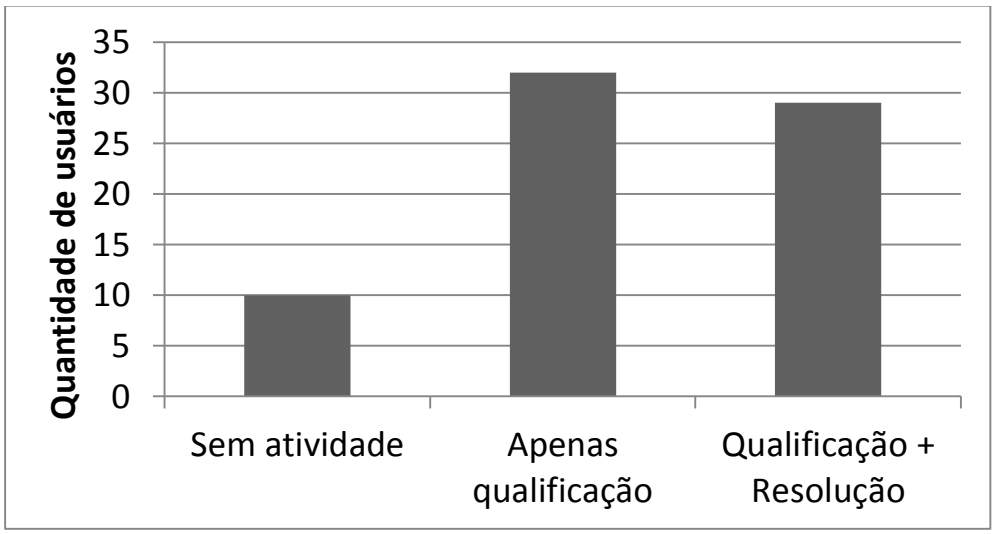


Figura 28 – Análise da distribuição de usuários por fases (elaborado pelo autor)

5.4.2.Comportamento na tarefa x número de ocorrências

O comportamento do usuário diante da tarefa se caracteriza pelas ações que este possuía durante a tarefa. A seguir as três possíveis ações dos usuários:

1) Tarefa completa – Quando o usuário finaliza a tarefa antes do tempo terminar.
2) Tempo esgotado – Quando o usuário realiza a tarefa até o tempo limite.
3) Desistência – Quando o usuário desiste por meio do botão de ação ou fecha a janela do navegador.

De acordo com a Figura 29, o número de usuários que completaram a tarefa na fase de resolução é muito maior do que os outros comportamentos, indicando assim que a maioria das pessoas estava compromissada com o experimento. A baixa ocorrência na fase qualificação é um indício de que esta fase era bastante fácil de ser ultrapassada.

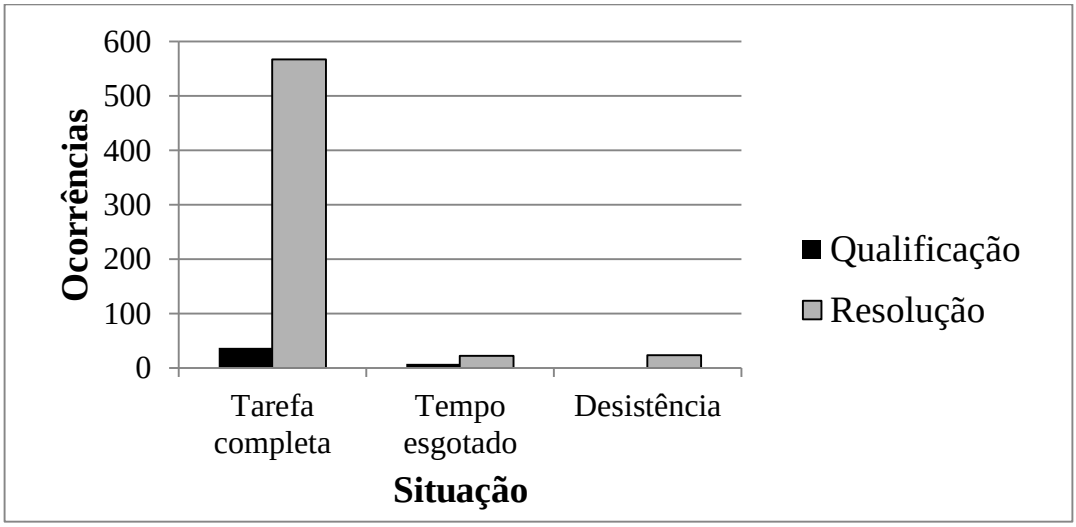


Figura 29 – Análise de comportamento dos usuários nas tarefas na fase de qualificação e resolução (elaborado pelo autor)

5.4.3.Tempo de execução x número de ocorrências

O tempo médio para a execução de uma tarefa na fase de qualificação foi de 85 segundos. Já para a fase de resolução o tempo médio foi de 51 segundos. Através da Figura 30, duas hipóteses podem ser levantadas para justificar essa diminuição. A primeira é que, na fase de qualificação, os usuários podem ter sido mais cautelosos, portanto, demoraram mais para executar uma tarefa. A segunda hipótese é que, conforme os usuários executavam a tarefa, aprimoravam cada vez mais suas habilidades nas tarefas, terminando-as em tempos menores.

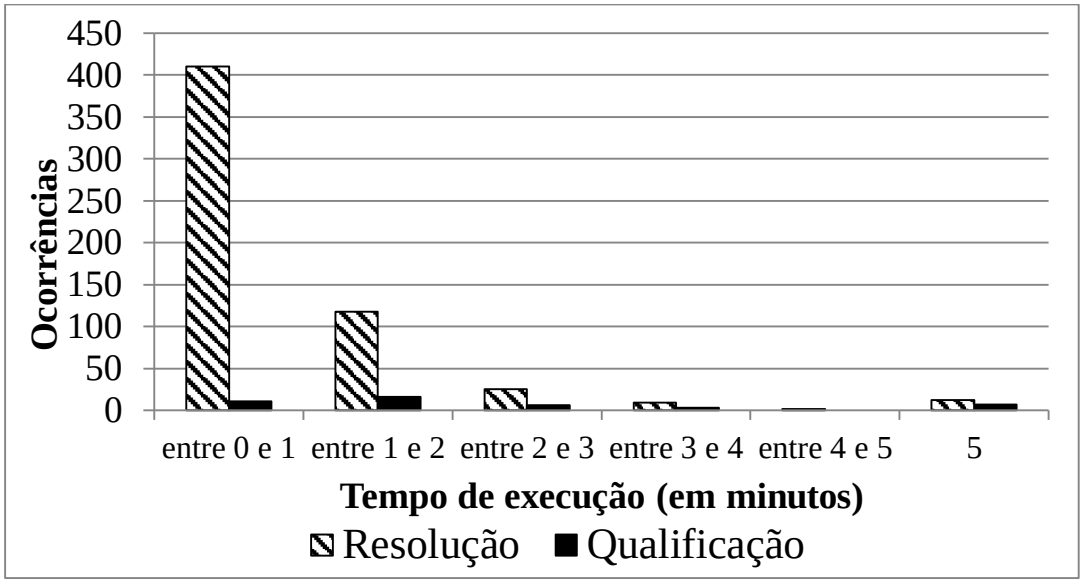


Figura 30 – Tempo de execução x ocorrências (elaborado pelo autor)

5.4.4. Usuários x ocorrência

A média de tarefas executadas na fase de qualificação foi de 1,34 tarefas por pessoa, enquanto que na fase de resolução foi de 21,1 tarefas por pessoa (Figura 31). Esses dados mostram que era relativamente fácil passar de fase de qualificação (60 pontos), o que indica uma possibilidade no aumento da pontuação mínima para qualificar ainda melhor a multidão.

Para a fase de resolução nota-se uma predominância no intervalo de 0 a 10 execuções (Figura 32), o que indica que a maioria dos usuários não se sentiram tão motivados para continuarem a resolver as tarefas, indicando que uma premiação ou monetização, possivelmente, poderia alavancar o número de tarefas por usuário.

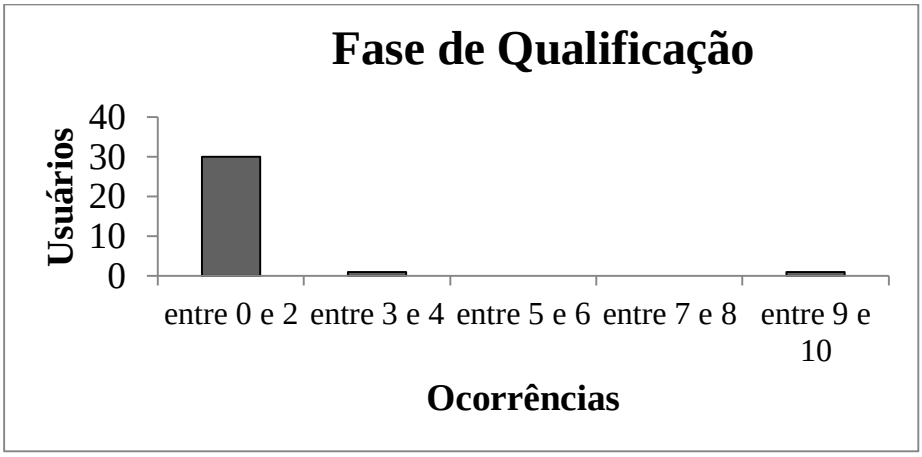


Figura 31 – Número de ocorrências necessárias para os usuários passarem da fase de qualificação (elaborado pelo autor)

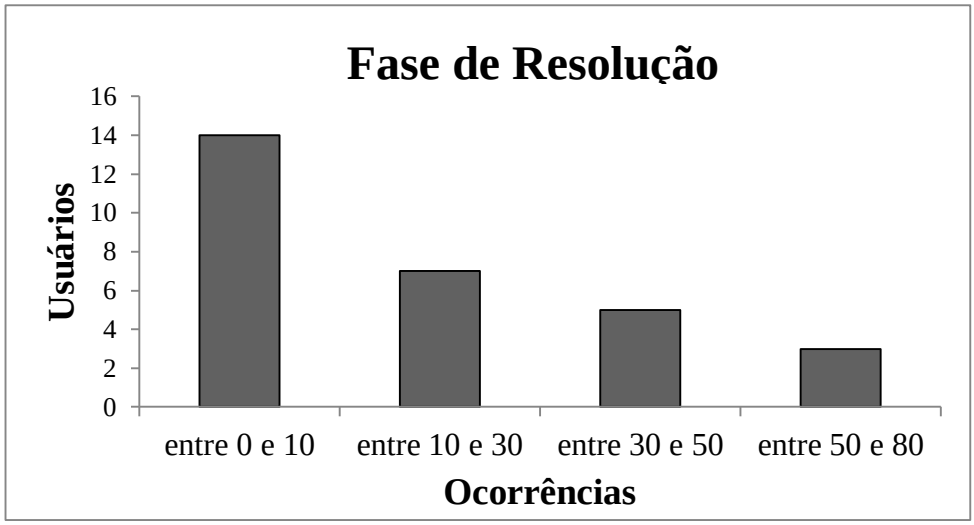


Figura 32 – Número de ocorrências dos usuários na fase de resolução (elaborado pelo autor)

5.4.5.Cálculo de Similaridade entre os resultados da multidão e gabarito

Após o término do experimento, todos os resultados dos usuários foram computados de modo a gerar os agrupamentos das referências. O algoritmo aplicado para gerar os agrupamentos foi o descrito no Apêndice 6.

A Tabela 17 apresenta os valores das métricas em função de um *threshold* informado manualmente. A variação desta métrica nos permite analisar com maior precisão os resultados gerados pela multidão.

O valor ótimo alcançado pelo sistema ocorre quando o *threshold* está configurado para 95%, com um grau de similaridade também de 95%. Como o CrowdER e o Corleone não utilizam o T-W *Index* em seus experimentos, a comparação deverá ser feita por *precision* e *recall*.

Tabela 17 – Análise de métricas aplicadas ao resultado da resolução de entidade executada pela multidão (elaborado pelo autor)

Threshold	**TRUE**	**0,40**	**0,5**	**0,6**	**0,7**	**0,8**	**0,9**	**0,95**
T-W Index	1	0,73	0,85	0,91	0,91	0,91	0,91	0,95
Recall	1	1	1	1	1	1	1	0,99
Precision	1	0,59	0,75	0,86	0,86	0,86	0,86	0,96
F1-Score	1	0,75	0,86	0,92	0,92	0,92	0,92	0,98
Número de agrupamentos	27	23	24	26	26	26	26	28
Tamanho Médio de um grupo	2,85	3,65	3,33	3,04	3,04	3,04	3,04	2,86
Distribuição das classes								
1	0	0	0	0	0	0	0	0
2	12	6	7	11	11	11	11	11
3	10	8	9	8	8	8	8	13
4	2	2	2	3	3	3	3	1
5	3	5	5	3	3	3	3	3
6	0	0	1	1	1	1	1	0
7	0	1	0	0	0	0	0	0
8+	0	0	0	0	0	0	0	0

O CrowdER (WANG *et al.*, 2012) tem o seu melhor desempenho com um *recall* de 92% (base de dados Produto), enquanto Corleone (GOKHALE *et al* , 2014) tem com 96% (base de dados Restaurante). A ferramenta CERM apresentou um recall de 100%. Devemos observar que essa situação ocorreu pelo fato do número de referências ser relativamente pequeno.

Analisando o CERM pela métrica *precision*, é possível notar uma melhoria de 75% para 86% quando o *threshold* é configurado de 40% a 50%.

Na Figura 33, é possível, de forma mais clara, visualizar a evolução dos valores das métricas. O recall sempre tem um valor máximo, mesmo para um *threshold* bem baixo. Isso mostra que a multidão tem uma alta qualidade, e que a maioria das resoluções executadas obtiveram êxito.

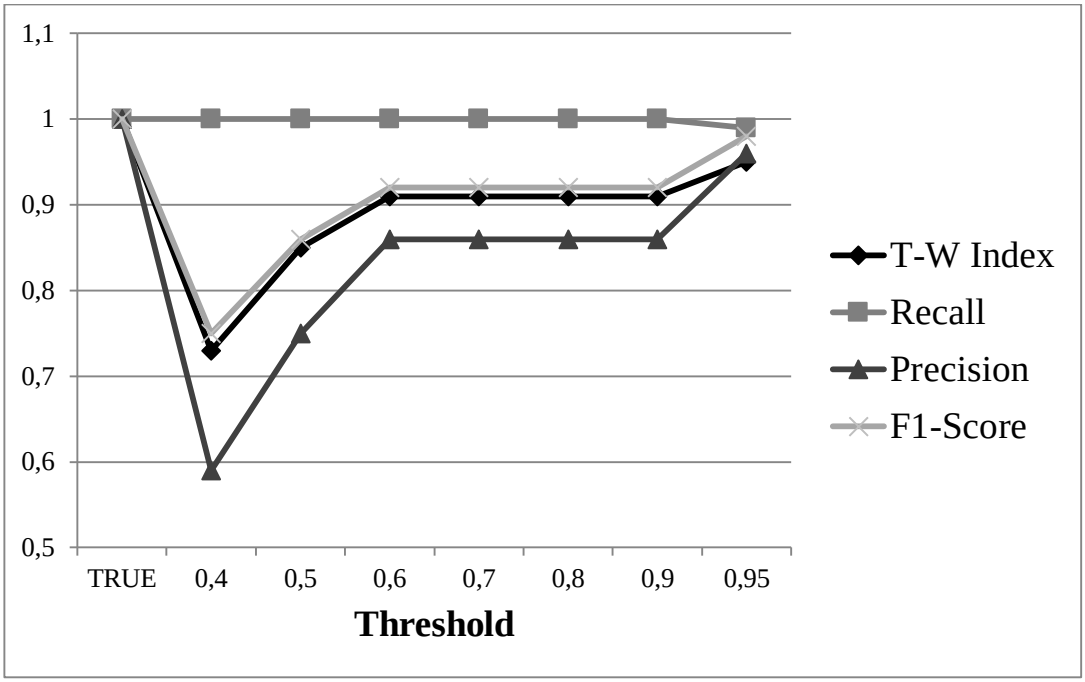


Figura 33 – Distribuição das métricas T-W Index, *precision, recall, F1-Score* em função do *threshold* (elaborado pelo autor)

A Figura 34, mostra que o aumento do *threshold* implica no aumento do número de clusters. Isso é totalmente compreensível, uma vez que os critérios utilizados para agrupar duas referências se tornam cada vez mais restritos.

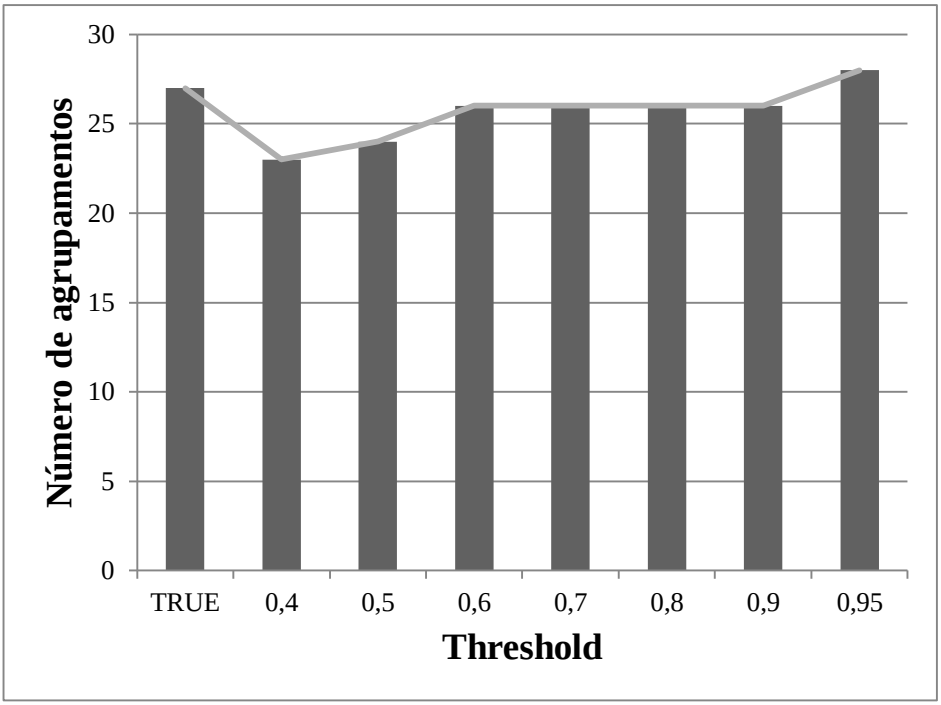


Figura 34 – Número de agrupamentos em função do *threshold* (elaborado pelo autor)

5.5.Perfil e respostas dos testadores

Os participantes preencheram um formulário sobre o nível de conhecimento teórico em *crowdsourcing*, mais especificamente, nas áreas de *Marketplace* e de *CrowdScience*.

O conhecimento em banco de dados, que a princípio não tem relação direta com a resolução de tarefas, mas que pode influenciar nos resultados, também foi considerado. O formato de apresentação de dados, tanto de pesquisa no filtro, lembra muito o conhecimento adquirido com modelagem de dados.

O formulário estava disponível no momento do cadastro na ferramenta CERM, onde as respostas estão no Apêndice 7. Esta seção dedica-se a estudar também as respostas fornecidas pelos participantes após o experimento. O formulário em questão encontra-se no Apêndice 8, e as respostas de cada usuário podem ser visualizadas no Apêndice 9.

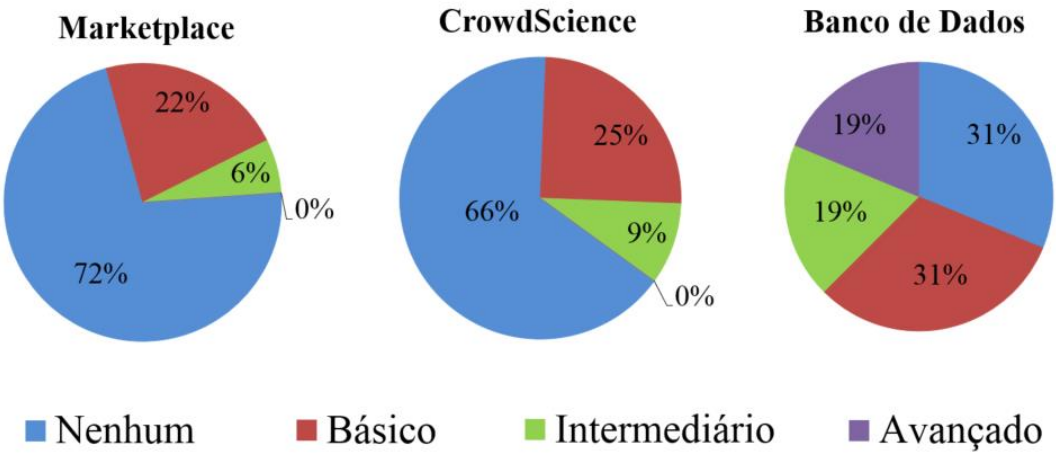


Figura 35 – Distribuição de participantes quanto ao conhecimento em *crowdsourcing* e banco de dados (elaborado pelo autor)

Como mostrado na Figura 35, ao todo, 32 usuários participaram da rodada de experimento. Nenhuma pessoa respondeu possuir conhecimento avançado em *Marketplace* e *CrowdScience*. A grande maioria dos usuários não tinha conhecimento a respeito de sistemas *crowdsourcing*.

5.6.Considerações Gerais

Outro fator analisado durante o experimento foi em relação à motivação dos usuários. Para aumentar a interação com a multidão, foi utilizado o recurso de *drag and drop*. Mesmo com esse recurso bem intuitivo, alguns usuários mostraram desinteresse no uso da ferramenta, pois não acharam a tarefa divertida. Um fator motivacional bem construído influencia de maneira positiva atraindo a multidão para a execução das tarefas.

O modelo criado se diferencia principalmente dos outros devido as suas estratégias de controle de qualidade. Além disso, a flexibilização da entrada de dados é uma enorme vantagem, frente as abordagens mais atuais que sempre assumem duas listas para resolução de entidade.

O grau de similaridade T-W Index obtido pela ferramenta CERM foi de 91%. Isto fortalece o argumento que a multidão tem capacidade para resolver os mais diferenciados tipos de entradas. Além disso, a ferramenta CERM obteve 86% de precisão em seus resultados.

Na Tabela 18 são apresentadas as características, de forma resumida, do modelo proposto por este estudo e desenvolvido com o nome de CERM comparando-o com as ferramentas previamente estudados.

Tabela 18 - Características das ferramentas de resolução de entidades com *crowdsourcing*, incluindo a ferramenta CERM (elaborado pelo autor)

Ferramenta	Etapa de ER	HOC	Dados de Entrada	Estratégias de Qualidade	Quem pode testar	HIT	Motivação da Multidão
Corleone	ERA3	Sim	Duas bases e atributos variados	-	Usuários do MTurk	Múltipla escolha (Sim, Não, Em dúvida)	Monetização por tarefa realizada
CrowdER	ERA3	Não	Duas bases e atributos variados	Único HIT para qualificação	Usuários do MTurk	Múltiplas escolhas por par / categorização múltipla	Monetização por tarefa realizada
CrowdMatchER	ERA3 (*schema*)	Não	Diversos *schemas* e atributos variados	-	Usuários do MTurk	Múltipla Escolha (Sim ou Não)	Monetização por tarefa realizada
ZenCrowd	ERA3	Não	Website	- Qualificação - Atualização de qualidade	Usuários do MTurk	Múltipla Escolha	Monetização por tarefa correta
CERM	ERA3	Sim	Diversas bases e atributos variados	- Qualificação - Atualização de qualidade - Perfil de Usuário	Qualquer um	Filtro para consulta de dados, *drag and drop*, tempo limite	Competição, diversão, altruísmo

6.Conclusão e Trabalhos Futuros

O grande crescimento no volume de dados na Web, bem como a popularização de atividades extremamente dependentes de dados, tais como: integração dos dados, mineração dos dados, *business intelligence*, fazem com que se dê cada vez mais importância à qualidade desses dados.

Dentre os problemas que afetam a qualidade dos dados, a duplicidade de registros ocorre de maneira frequente. Resolução de Entidades tem como objetivo primário identificar referências que indicam o mesmo objeto no mundo real. A forma mais comum para executar o processo de busca às referências duplicadas ocorre através de algoritmos computacionais otimizados para tal tarefa.

Os algoritmos estão longe da perfeição quando o assunto é tratar referências que representam o mesmo objeto do mundo real. É nesse momento que a sabedoria da multidão pode auxiliar nessas inúmeras possibilidades de comparação (SUROWIECKI, 2006, LEVY, 2007). Adotando o conceito de computação humana (VON AHN, 2005), apenas o poder de processamento humano é capaz de resolver determinadas tarefas.

A proposta deste trabalho não é provar que a abordagem puramente *crowdsourcing* é melhor ou pior do que uma abordagem híbrida. Entretanto, o objetivo principal é mostrar que é possível, sim, utilizar a capacidade da multidão para resolver problemas de resolução de entidades de qualquer gênero.

Este trabalho utilizou três bases de dados com diferentes atributos, inclusive com imagens dos produtos mostrando que a resolução pode envolver diferentes informações. Vídeos e áudios também poderiam ser usados na ferramenta CERM, necessitando apenas de pequenas adaptações.

As conclusões que podem ser analisadas deste trabalho é que a multidão pode realizar tarefas de resolução de entidades com um grau maior de liberdade sem perder a qualidade dos dados. Para isso foi necessário a utilização de estratégias que pudessem auxiliar na separação de usuários com baixa qualidade daqueles com excelente qualidade.

A seguir, os trabalhos futuros que podem ser gerados a partir deste trabalho:

- Incluir a atividade ERA4 (seção 3.4.4) ao modelo pode ser uma forma de reduzir esforço, custo e tempo, pois o modelo contemplaria o gerenciamento de identidade das entidades. Logo, ao acrescentar uma nova base de dados, não seria necessário refazer toda a resolução

permitindo que a resolução realizada anteriormente de uma entidade seja reaproveitada em novas resoluções da mesma entidade, porém com outras bases ou dados. Os agrupamentos já formados podem substituir o lugar da referência alvo, por exemplo. Porém, ainda seriam necessárias outras adaptações.

- Um problema na resolução de entidades é que, a priori, é uma tarefa cansativa e chata, criar e desenvolver um ambiente onde os elementos de gamificação (*bagdes*, *achievements*) estejam presentes para motivar os usuários, tornando a tarefa em algo casual, simples e divertido, são medidas que podem fidelizar a utilização da ferramenta, podendo aumentar a qualidade dos resultados a longo prazo.
- Expansão do modelo contemplando níveis hierárquicos, onde até a própria multidão pode ser usada para gerar o gabarito para a fase de qualificação.
- Assim como a ferramenta ZenCrowd, o modelo proposto pode também realizar resolução de entidades para dados ligados, explorando o potencial da multidão com maior ênfase do que as abordagens atuais.
- A primeira atividade de resolução de entidades (ERA 1) também pode ser resolvida com o modelo proposto, onde a multidão teria maior participação para estruturar as informações, refinando os dados para que a etapa ERA 3 apresente melhores resultados.

Referências Bibliográficas

VON AHN, L., 2005. *Human Computation*. . Phd. Pittsburgh, Pennsylvania: Carnegie Mellon University.

VON AHN, L., 2009. "Human Computation". In: *Proceedings of the 46th Annual Design Automation Conference*. New York, NY, USA: ACM. 2009. pp. 418–419.

VON AHN, L., BLUM, M., HOPPER, N., et al., 2003. "CAPTCHA: Using Hard AI Problems for Security". In: BIHAM, Eli (ed.), *Advances in Cryptology — EUROCRYPT 2003*. S.l.: Springer Berlin / Heidelberg. pp. 646–646.

VON AHN, L., DABBISH, L., 2004. "Labeling Images with a Computer Game". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. New York, NY, USA: ACM. 2004. pp. 319–326.

VON AHN, L., MAURER, B., MCMILLEN, C., et al., 2008, "reCAPTCHA: Human-Based Character Recognition via Web Security Measures". In: *Science*. v. 321, pp. 1465–1468.

ALONSO, O., BAEZA-YATES, R., 2011. "Design and Implementation of Relevance Assessments Using Crowdsourcing". In: CLOUGH, Paul, FOLEY, Colum, GURRIN, Cathal, JONES, Gareth J. F., KRAAIJ, Wessel, LEE, Hyowon & MUDOCH, Vanessa (eds.), *Advances in Information Retrieval*. S.l.: Springer Berlin Heidelberg. Lecture Notes in Computer Science, 6611. pp. 153–164.

ALT, H., SCHARF, L., SCHOLZ, S., 2007. "Probabilistic Matching and Resemblance Evaluation of Shapes in Trademark Images". In: *Proceedings of the 6th ACM International Conference on Image and Video Retrieval*. New York, NY, USA: ACM. 2007. pp. 533–540.

ARKADY, M., 2007, *Data Quality Assessment*. . USA, Technics Publications, LLC.

BENJELLOUN, O., GARCIA-MOLINA, H., MENESTRINA, D., et al., 2009, "Swoosh: A Generic Approach to Entity Resolution". In: *The VLDB Journal*. v. 18, pp. 255–276.

BILGIC, M., LICAMELE, L., GETOOR, L., et al., 2006. "D-Dupe: An Interactive Tool for Entity Resolution in Social Networks". In: *Visual Analytics Science And Technology, 2006 IEEE Symposium On*. S.l.: s.n. Outubro 2006. pp. 43–50.

BRABHAM, D.C., 2007. Disponível em: <http://crowdsourcing.typepad.com/cs/2007/04/speakers_corner.html>. Acessado em: 22 Agosto 2014.

BRABHAM, D.C., 2008, "Crowdsourcing as a Model for Problem Solving An Introduction and Cases". In: *Convergence: The International Journal of Research into New Media Technologies*. v. 14, pp. 75–90.

CHAN, Y., TALBURT, J., TALLEY, T.M., 2009, *Data Engineering: Mining, Information and Intelligence*. . 2010 edition. New York, Springer.

CHEN, D.L., DOLAN, W.B., 2011, *Building a Persistent Workforce on Mechanical Turk for Multilingual Data Collection*. . S.l., s.n.

CHEN, Z., KALASHNIKOV, D.V., MEHROTRA, S., 2009. "Exploiting Context Analysis for Combining Multiple Entity Resolution Systems". In: *Proceedings of the 2009 ACM SIGMOD International Conference on Management of Data*. New York, NY, USA: ACM. 2009. pp. 207–218.

CHIANG, C.-C., TALBURT, J., WU, N., et al., 2008. "A Case Study in Partial Parsing Unstructured Text". In: *Proceedings of the Fifth International Conference on Information Technology: New Generations*. Washington, DC, USA: IEEE Computer Society. 2008. pp. 447–452.

DEMARTINI, G., DIFALLAH, D.E., CUDRÉ-MAUROUX, P., 2012. "ZenCrowd: Leveraging Probabilistic Reasoning and Crowdsourcing Techniques for Large-scale Entity Linking". In: *Proceedings of the 21st International Conference on World Wide Web*. New York, NY, USA: ACM. 2012. pp. 469–478.

DOWNS, J.S., HOLBROOK, M.B., SHENG, S., et al., 2010. "Are Your Participants Gaming the System?: Screening Mechanical Turk Workers". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. New York, NY, USA: ACM. 2010. pp. 2399–2402.

DYBA, T., DINGSOYR, T., HANSSEN, G.K., 2007. "Applying Systematic Reviews to Diverse Study Types: An Experience Report". In: *2013 ACM / IEEE International Symposium on Empirical Software Engineering and Measurement*. Los Alamitos, CA, USA: IEEE Computer Society. 2007. pp. 225–234.

FELLEGI, I.P., SUNTER, A.B., 1969, "A Theory for Record Linkage". In: *Journal of The American Statistical Association*. v. 64, pp. 1183–1210.

FORTSON, L., MASTERS, K., NICHOL, R., et al., 2011, "Galaxy Zoo: Morphological Classification and Citizen Science". In: *arXiv:1104.5513 [astro-ph]*.

GALAXYZOO, 2014. http://www.galaxyzoo.org/.

GARCIA-MOLINA, H., 2006. "Pair-Wise Entity Resolution: Overview and Challenges". In: *Proceedings of the 15th ACM International Conference on Information and Knowledge Management*. New York, NY, USA: ACM. 2006. pp. 1–1.

GEIGER, D., SEEDORF, S., SCHULZE, T., et al., 2011, "Managing the Crowd: Towards a Taxonomy of Crowdsourcing Processes". In: *AMCIS 2011 Proceedings - All Submissions*.

GOKHALE, C., DAS, S., DOAN, A., et al., 2014. "Corleone: Hands-off Crowdsourcing for Entity Matching". In: *Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data*. New York, NY, USA: ACM. 2014. pp. 601–612.

GOUTTE, C., GAUSSIER, E., 2005. "A Probabilistic Interpretation of Precision, Recall and F-score, with Implication for Evaluation". In: *Proceedings of the 27th European Conference on Advances in Information Retrieval Research*. Berlin, Heidelberg: Springer-Verlag. 2005. pp. 345–359.

HERNÁNDEZ, M.A., STOLFO, S.J., 1995. "The Merge/Purge Problem for Large Databases". In: *Proceedings of the 1995 ACM SIGMOD International Conference on Management of Data*. New York, NY, USA: ACM. 1995. pp. 127–138.

HERZOG, T.N., SCHEUREN, F.J., WINKLER, W.E., 2007, *Data Quality and Record Linkage Techniques*. . 1st. S.l., Springer Publishing Company, Incorporated.

HOWE, J., 2006a. Disponível em: <http://crowdsourcing.typepad.com/cs/2006/06/crowdsourcing_a.html>. Acessado em: 22 Julho 2014.

HOWE, J., 2006b. Disponível em: <http://archive.wired.com/wired/archive/14.06/crowds.html>. Acessado em: 1 Julho 2014.

HOWE, J., 2008, *Crowdsourcing: Why the Power of the Crowd Is Driving the Future of Business*. . S.l., Crown Publishing Group.

HOWE, J., 2009, *Poder Das Multidoes, O*. . Tradução por: ALESSANDRA MUSSI ARAUJO. S.l., Elsevier Brasil.

INMON, W.H., NESAVICH, A., 2007, *Tapping into Unstructured Data: Integrating Unstructured Data and Textual Analytics into Business Intelligence*. . 1 edition. Upper Saddle River, NJ, Prentice Hall.

IPEIROTIS, P.G., PROVOST, F., WANG, J., 2010. "Quality Management on Amazon Mechanical Turk". In: *Proceedings of the ACM SIGKDD Workshop on Human Computation*. New York, NY, USA: ACM. 2010. pp. 64–67.

JACCARD, P., 1901, "Étude comparative de la distribution florale dans une portion des Alpes et des Jura". In: *Bulletin del la Société Vaudoise des Sciences Naturelles*. v. 37, pp. 547–579.

JARO, M.A., 1989, "Advances in Record-Linkage Methodology as Applied to Matching the 1985 Census of Tampa, Florida". In: *Journal of the American Statistical Association*. v. 84, pp. 414–420.

KAZAI, G., 2011. "In Search of Quality in Crowdsourcing for Search Engine Evaluation". In: *Proceedings of the 33rd European Conference on Advances in Information Retrieval*. Berlin, Heidelberg: Springer-Verlag. 2011. pp. 165–176.

KAZAI, G., KAMPS, J., MILIC-FRAYLING, N., 2011. "Worker Types and Personality Traits in Crowdsourcing Relevance Labels". In: *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*. New York, NY, USA: ACM. 2011. pp. 1941–1944.

KITCHENHAM, B., CHARTERS, S., 2007. *Guidelines for performing Systematic Literature Reviews in Software Engineering*. S.l. Acessado em: 7 Julho 2013. Disponível em: <http://www.dur.ac.uk/ebse/resources/Systematic-reviews-5-8.pdf>.

KITTUR, A., CHI, E.H., SUH, B., 2008. "Crowdsourcing User Studies with Mechanical Turk". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. New York, NY, USA: ACM. 2008. pp. 453–456.

KLEIN, D., MANNING, C.D., 2003. "Accurate Unlexicalized Parsing". In: *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1*. Stroudsburg, PA, USA: Association for Computational Linguistics. 2003. pp. 423–430.

KÖPCKE, H., RAHM, E., 2010, "Frameworks for entity matching: A comparison". In: *Data & Knowledge Engineering*. v. 69, pp. 197–210.

LE, J., EDMONDS, A., HESTER, V., et al., 2010. "Ensuring quality in crowdsourced search relevance evaluation: The effects of training question distribution". In: *Proc. SIGIR 2010 Workshop on Crowdsourcing for Search Evaluation*. S.l.: s.n. 2010. pp. 21–26.

LEVENSHTEIN, V., 1966, "Binary Codes Capable of Correcting Deletions, Insertions, and Reversals". In: *Soviet Physics-Doklady*. v. 10, pp. 707–710.

LÉVY, P., 1998, *A inteligência coletiva: por uma antropologia do ciberespaço*. . Tradução por: Luiz Paulo Rouanet. 5 ed. São Paulo, Brasil, Edições Loyola.

LEVY, P., 2007, *Inteligência coletiva (A)*. . S.l., Edicoes Loyola.

LIM, E.-P., SRIVASTAVA, J., PRABHAKAR, S., et al., 1993. "Entity Identification in Database Integration". In: *Proceedings of the Ninth International Conference on Data Engineering*. Washington, DC, USA: IEEE Computer Society. 1993. pp. 294–301.

LOSHIN, D., 2008, *Master Data Management*. . San Francisco, CA, USA, Morgan Kaufmann Publishers Inc.

MALIN, B., SWEENEY, L., 2005, *ENRES: A Semantic Framework for Entity Resolution Modelling*. . S.l., s.n.

MARCH, J.G., 1991, "Exploration and Exploitation in Organizational Learning". In: *Organization Science*. v. 2, pp. 71–87.

MENESTRINA, D., BENJELLOUN, O., GARCIA-MOLINA, H., 2005. Disponível em: <http://ilpubs.stanford.edu:8090/699/>. Acessado em: 17 Setembro 2014.

MITCHELL, D.W., 2004, "More on Spreads and Non-Arithmetic Means". In: *The Mathematical Gazette*. v. 88, pp. 142–144.

NAUMANN, F., HERSCHEL, M., 2010, *An Introduction to Duplicate Detection*. . S.l., Morgan & Claypool Publishers.

NEWCOMBE, H.B., KENNEDY, J.M., AXFORD, S.J., et al., 1959, "Automatic Linkage of Vital Records Computers". In: *Science*. v. 130, pp. 954–959.

NOVOTNEY, S., CALLISON-BURCH, C., 2010. "Shared Task: Crowdsourced Accessibility Elicitation of Wikipedia Articles". In: *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*. Stroudsburg, PA, USA: Association for Computational Linguistics. 2010. pp. 41–44.

PAGE, S.E., 2007, *The Difference: How the Power of Diversity Creates Better Groups, Firms, Schools, and Societies*. . S.l., Princeton University Press.

PINKERTON, B., 1994. "Finding What People Want: Experiences with the WebCrawler". In: *First World Wide Web Conference*. Geneva, Switzerland: s.n. 1994.

QUINN, A.J., BEDERSON, B.B., 2009, "A taxonomy of distributed human computation". In: *Human-Computer Interaction Lab Tech Report, University of Maryland*.

QUINN, A.J., BEDERSON, B.B., 2011. "Human computation: a survey and taxonomy of a growing field". In: *Proceedings of the 2011 annual conference on Human factors in computing systems*. New York, NY, USA: ACM. 2011. pp. 1403–1412.

RAYKAR, V.C., YU, S., ZHAO, L.H., et al., 2010, "Learning From Crowds". In: *J. Mach. Learn. Res.* v. 11, pp. 1297–1322.

REDMAN, T.C., 2008, *Data Driven: Profiting from Your Most Important Business Asset*. . S.l., Harvard Business Press.

ROSS, J., IRANI, L., SILBERMAN, M.S., et al., 2010. "Who are the crowdworkers?: shifting demographics in Mechanical Turk". In: *In Proceedings of CHI 2010, Atlanta GA, ACM*. S.l.: s.n. 2010.

SAHA, R., MANNA, R., GEETHA, G., 2012. "CAPTCHINO - A Gamification of Image-Based CAPTCHAs to Evaluate Usability Issues". In: *2012 International Conference on Computing Sciences (ICCS)*. S.l.: s.n. Setembro 2012. pp. 95–99.

SAS POSITIONED AS A LEADER IN MAGIC QUADRANT FOR DATA QUALITY TOOLS | SAS PRESS RELEASES, 2014. http://www.sas.com/news/preleases/data-quality-gartnermq.html.

SCHENK, E., GUITTARD, C., 2011, "Towards a characterization of crowdsourcing practices". In: *Journal of Innovation Economics*. v. 7, pp. 93.

SNOW, R., O'CONNOR, B., JURAFSKY, D., et al., 2008. "Cheap and Fast—but is It Good?: Evaluating Non-expert Annotations for Natural Language Tasks". In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Stroudsburg, PA, USA: Association for Computational Linguistics. 2008. pp. 254–263.

SUROWIECKI, J., 2005, *The Wisdom of Crowds*. . S.l., Knopf Doubleday Publishing Group.

SUROWIECKI, J., 2006, *A Sabedoria da Multidões*. . Tradução por: Alexandre Martins. Rio de Janeiro, Record.

TALBURT, J.R., 2010, *Entity Resolution and Information Quality*. . 1st. San Francisco, CA, USA, Morgan Kaufmann Publishers Inc.

TALBURT, J.R., ZHOU, Y., SHIVAIAH, S.Y., 2009. "SOG: A Synthetic Occupancy Generator to Support Entity Resolution Instruction and Research". In: BOWEN, Paul L., ELMAGARMID, Ahmed K., ÖSTERLE, Hubert & SATTLER, Kai-Uwe (eds.), *Proceedings of the 14th International Conference on Information Quality, ICIQ 2009, Hasso Plattner Institute, University of Potsdam, Germany, November 7-8 2009*. S.l.: HPI/MIT. 2009. pp. 91–105.

TALBURT, J., WANG, R., HESS, K., et al., 2007, *Information Quality Management: Theory and Applications*. . S.l., Hershey. Acessado em: 8 Agosto 2014.

TANSEL, B., BRIZAN, D.G., TANSEL, A.U., 2006, "A Survey of Entity Resolution and Record Linkage Methodologies". In: *Communications of the IIMA*. pp. 41–50.

UKKONEN, E., 1992, "Approximate String-matching with Q-grams and Maximal Matches". In: *Theor. Comput. Sci.* v. 92, pp. 191–211.

VENHUIZEN, N.J., BASILE, V., EVANG, K., et al., 2013, "Gamification for Word Sense Labeling". In: *Proceedings of the 10th International Conference on Computational Semantics (IWCS 2013) – Short Papers*. pp. 397–403.

VUURENS, J., P. DE VRIES, A., EICKHOFF, C., 2011. "How Much Spam Can You Take? An Analysis of Crowdsourcing Results to Increase Accuracy". In: *ACM SIGIR Workshop on Crowdsourcing for Information Retrieval (CIR'11)*. S.l.: ACM. 2011. pp. 21–26.

WAIS, P., LINGAMNENI, S., COOK, D., et al., 2010. "Towards Building a High-Quality Workforce with Mechanical Turk". In: *NIPS Workshop on Computational Social Science and the Wisdom of Crowds*. S.l.: s.n. Dezembro 2010.

WANG, J., KRASKA, T., FRANKLIN, M.J., et al., 2012, "Crowder: Crowdsourcing entity resolution". In: *PVLDB*. pp. 59.

WANG, R.Y.-Y., STRONG, D.M., 1996, "Beyond accuracy: what data quality means to data consumers". In: *Journal of Management Information Systems*. v. 12, pp. 5–33.

WATTS, D.J., STROGATZ, S.H., 1998, "Collective dynamics of "small-world" networks". In: *Nature*. v. 393, pp. 440–442.

WEBSTER, J., WATSON, R.T., 2002, "Analyzing the Past to Prepare for the Future: Writing a Literature Review". In: *MIS Q.* v. 26, pp. xiii–xxiii.

WHANG, S.E., LOFGREN, P., GARCIA-MOLINA, H., 2013, "Question Selection for Crowd Entity Resolution". In: *Proc. VLDB Endow.* v. 6, pp. 349–360.

WINKLER, W.E., 1999. *The State of Record Linkage and Current Research Problems*. S.l. Statistical Research Division, U.S. Census Bureau.

WINKLER, W.E., 2006. *Overview of record linkage and current research directions*. S.l. BUREAU OF THE CENSUS.

WU, N., TALBURT, J., HEIEN, C., et al., 2007, "A Method for Entity Identification in Open Source Documents with Partially Redacted Attributes". In: *J. Comput. Sci. Coll.* v. 22, pp. 138–144.

YUEN, M.-C., KING, I., LEUNG, K.-S., 2011. "A Survey of Crowdsourcing Systems". In: *2011 IEEE Third International Conference on Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third Inernational Conference on Social Computing (SocialCom)*. S.l.: s.n. Outubro 2011. pp. 766–773.

ZHANG, C.J., CHEN, L., JAGADISH, H.V., et al., 2013, "Reducing Uncertainty of Schema Matching via Crowdsourcing". In: *Proc. VLDB Endow.* v. 6, pp. 757–768.

ZHANG, C.J., ZHAO, Z., CHEN, L., et al., 2014. "CrowdMatcher: Crowd-assisted Schema Matching". In: *Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data*. New York, NY, USA: ACM. 2014. pp. 721–724.

ZHOU, Y., NELSON, E., KOBAYASHI, F., et al., 2013, "A Graduate-Level Course on Entity Resolution and Information Quality: A Step Toward ER Education". In: *J. Data and Information Quality*. v. 4, pp. 10:1–10:10.

ZHOU, Y., TALBURT, J., 2011, "Staging a Realistic Entity Resolution Challenge for Students". In: *J. Comput. Sci. Coll.* v. 26, pp. 88–95.

ZHU, D., CARTERETTE, B., 2011. "An analysis of assessor behavior in crowdsourced preference judgments". In: . S.l.: s.n. 2011.

2014. http://www.wikipedia.org/.

2014. http://www.youtube.com/.

2014. http://www.w3.org/XML/.

2014. http://www.infoglide.com/technology/identity-resolution-engine-ire/.

Apêndices

No Apêndice 1 é apresentado a revisão sistemática com informações detalhadas sobre a pesquisa, os termos utilizados, as conferências selecionadas, os critérios de avaliações adotados.

Os Apêndice 2 e 3 apresentam os modelos entidade-relacionamento criado para a ferramenta CERM. O diagrama de classe e de caso de uso são apresentados nos Apêndices 4 e 5, respectivamente.

O Apêndice 6 apresenta o algoritmo utilizado na ferramenta CERM para calcular o gabarito final dos resultados gerados pela multidão.

O Apêndice 7 apresenta o perfil dos participantes dos experimentos.

O formulário enviado aos usuários da ferramenta CERM após a utilização da mesma é apresentado no Apêndice 8. E as respostas dos formulários são exibidas no Apêndice 9.

Apêndice 1 - Revisão Sistemática sobre *Entity Resolution e Crowdsourcing*

A revisão sistemática de literatura foi utilizada para a elaboração da pesquisa de trabalhos relacionados. A escolha deste método se deve, principalmente, ao seu rigor na busca de trabalhos, e pela facilidade na identificação de lacunas em determinada área (KITCHENHAM & CHARTERS, 2007). A motivação por este método era encontrar trabalhos acadêmicos que relacionasse *entity resolution* com *crowdsourcing*.

DYBA *et al.* (2007) apresentam a revisão sistemática da literatura como uma forma de estudo secundário caracterizado por ser rigoroso para identificar, avaliar e resumir os estudos sobre o assunto.

KITCHENHAM & CHARTERS (2007) defendem que a revisão sistemática de literatura consiste em três fases:

1. Planejamento – Quando fontes, objetivos, critérios, e protocolo para execução são definidos.
2. Execução – Quando o protocolo definido é executado.
3. Relatório – Quando os dados dos artigos são analisados, extraídos e resumido.

A seguir, algumas diferenças entre a revisão tradicional e a sistemática são apresentadas:

1. Na revisão sistemática a questão é mais específica do que nos métodos tradicionais
2. A avaliação da revisão sistemática é mais rígida, enquanto em outros métodos é variável.
3. O resumo pode ser qualitativo ou quantitativo na revisão sistemática, porém nos métodos tradicionais é apenas qualitativo.
4. O critério de seleção é aplicado de forma uniforme na revisão sistemática, nos métodos tradicionais podem ser tendenciosas.
5. O esforço empregado na revisão sistemática pode ser relativamente alto.

Fases da Pesquisa

Ter conhecimento sobre a área de pesquisa é um importante passo para um trabalho bem fundamentado. A revisão sistemática da literatura foi realizada a fim de averiguar o estado das pesquisas direcionadas na aplicação de *crowdsourcing* em *entity resolution*. Esta seção tem como objetivo descrever a execução das três fases da revisão sistemática.

Fase 1 – Planejamento

O primeiro passo foi elaborar a questão que representa a motivação deste trabalho. A questão criada foi: "*Crowdsourcing* pode ser aplicado no contexto de ER?". E ainda mais, questões complementares foram formuladas: (1) Quais etapas de ER a multidão é capaz de resolver? (2) Quais habilidades são necessárias? (3) Qual a motivação desses usuários? Quais estratégias aumentam a acurácia da ER? Os artigos selecionados a partir da revisão sistemática responderam essas questões.

As seguintes bases foram selecionadas para realizar a busca: ACM Digital Library[51], CiteseerX[52], IEEE[53], Scopus[54], SciELO[55], ScienceDirect[56], Springer[57] e Web of Knowledge[58], observando que a busca foi sem restrição de ano. Além disso, cinco conferências foram manualmente examinadas, a partir do ano de 2008: CIKM (International Conference on Information and Knowledge Management), ICDE (International Conference on Data Engineering), ICUIMC (International Conference on Ubiquitous Information Management and Communication), ACM SIGMOD (Special Interest Group on Management of Data) Conference e VLDB (International Conference on Very Large Data Bases).

O critério para os estudos primários era a utilização de multidão na resolução de entidades onde era proposto um modelo, e para os estudos secundários a citação de

[51] http://dl.acm.org/

[52] http://citeseer.ist.psu.edu/

[53] http://ieeexplore.ieee.org/

[54] http://www.scopus.com/

[55] http://www.scielo.org/

[56] http://www.sciencedirect.com/

[57] http://link.springer.com/

[58] http://www.webofknowledge.com/

ferramentas que aplicam a resolução de entidades. Somente os trabalhos na língua inglesa foram considerados.

Os termos “entity resolution” e “crowdsourcing” e seus equivalentes, formaram a *string* de busca: “(((‘entity resolution’ OR ‘entity matching’ OR ‘entity identity’ OR ‘entity matching’ OR ‘entity linking’ OR ‘entity recognition’ OR ‘entity disambiguation’ OR ‘entity coereference’ OR ‘coreference resolution’ OR ‘instance matching’ OR ‘schema matching’ OR ‘matching pair’ OR ‘matching process’ OR ‘record linkage’ OR ‘deduplication’ OR ‘data matching’) AND (‘crowdsourcing’ OR ‘crowdsource’ OR ‘crowd computing’ OR ‘crowd-based’ OR ‘crowd based’ OR ‘crowding-based’ OR ‘wisdom of crowds’ OR ‘collective intelligence’ OR ‘crowd’ OR ‘crowdsourced’)) OR ‘crowd entity resolution’)”. Importante salientar que os termos entre aspas simples não podem ser desmembrados, a fim de maximizar o número de trabalhos que tem relação com as questões definidas.

Fase 2 – Execução

Cada base trabalha com uma forma de pesquisa avançada, portanto, oferece a possibilidade de aplicação de filtros mais específicos de modo que os resultados retornados sejam mais próximos do objetivo da revisão sistemática. O parâmetro “TITLE-ABS-KEY” realiza a busca somente nos campos de título, *abstract* e *meta tags*. Este parâmetro foi aplicado na base Scopus e ScienceDirect e reduziu a quantidade de trabalhos de 386 e 309 para 46 e 1 respectivamente.

A execução da busca nas bases foi realizada em 10 de janeiro de 2014. A Tabela 19 apresenta os dados separados por cada mecanismo de busca. Logo abaixo, estão passos e os resultados da execução:

1. A busca resultou em 69 estudos, sendo 18 repetições. Totalizando 51 estudos diferentes.
2. O título e o abstract dos trabalhos restantes foram analisados levando em conta as questões propostas pela revisão, o que reduziu o número para 25 trabalhos.
3. Nesse passo, a leitura da introdução foi analisada, caso não fosse compatível com a pesquisa, o mesmo era excluído.
4. Os trabalhos restantes eram lidos por completo baseando-se nas questões da revisão.

5. Finalmente, quatro estudos obedeceram aos critérios primário de inclusão. Nenhum artigo contemplou o segundo critério.

Tabela 19 - Resultado da busca por artigos da revisão sistemática (elaborado pelo autor)

	Journal	Conferência	Capítulo	Outro	Total
ACM	2	4	0	0	**6**
CiteseerX	0	4	0	0	**4**
IEEE	0	4	0	0	**4**
ScienceDirect	1	0	0	0	**1**
Scopus	4	31	8	3	**46**
SciELO	0	0	0	0	**0**
Springer	0	0	1	0	**1**
Web of Knowledge	1	4	2	0	**7**
Total	8	47	11	3	**69**
Não repetidos	6	34	6	5	**51**
Selecionado	4	19	2	0	**25**
Incluído	**0**	**4**	**0**	**0**	**4**

Dentre os trabalhos retornados, foram executados os passos *forward* e *backward* (WEBSTER & WATSON, 2002). *Backward* é o processo de recuperar artigos citados por um trabalho e verificar se os mesmos podem ser incluídos sob as restrições da revisão sistemática. *Forward* é o processo de recuperar os artigos que citaram o trabalho em questão. A funcionalidade "Citado por" Google Scholar foi utilizado para realizar o passo *Forward*. Estes procedimentos não resultaram em nenhum estudo, seja primário ou secundário.

Durante a leitura dos 25 trabalhos foi observado a presença de dois autores previamente já estudados, por exemplo, WANG *et al.* (2012) e GOKHALE *et al.* (2014).

Fase 3 – Relatório

Cada um dos estudos primários apresentou uma ferramenta que utiliza a multidão para a resolução de entidades: CrowdER (WANG *et al.*, 2012), ZenCrowd (DEMARTINI *et al.*, 2012), CrowdMatcher (ZHANG *et al.*, 2014) e Corleone (GOKHALE *et al.*, 2014).

A partir dos dados extraídos de cada uma das ferramentas de resolução de entidades, um comparativo foi elaborado. Através da pesquisa, foi possível responder as questões propostas inicialmente na revisão sistemática. A pergunta principal era: "*Crowdsourcing* pode ser aplicado no contexto de ER?". A resposta é sim. Especialmente as aplicações CrowdER e Corleone, que apresentaram o modelo mais completo para resolução de entidades. Embora, ambos modelos têm limitações principalmente no formato de entrada de dados. As outras questões e respostas são:

(1) Quais etapas de ER a multidão é capaz de resolver? Em geral, os resultados da pesquisa apontam para principalmente para o ERA 3, ou seja, a resolução de entidade propriamente dita. Não foram encontrados modelos que cobrissem todas as etapas de ER utilizando *crowdsourcing*. (2) Quais habilidades são necessárias? As abordagens adotadas pela maioria dos trabalhos eram em geral, questões de múltiplas escolhas, ou agrupamentos, tarefas simples para a multidão, então não eram necessárias habilidades conhecidas. (3) Qual a motivação desses usuários? Todos os modelos retornados pela revisão sistemática utilizavam a base de usuários do AMT para a realização das tarefas, então a motivação principal foi a recompensa monetária. (4) Quais estratégias aumentam a acurácia da ER? Avaliação do trabalhador, remoção de trabalhadores de baixa qualidade, rodada de qualificação, peso nas respostas dos trabalhadores de alta qualidade, utilização de acurácia do trabalhador fornecido pelo AMT.

Apêndice 2 - Modelo de Dados da ferramenta CERM

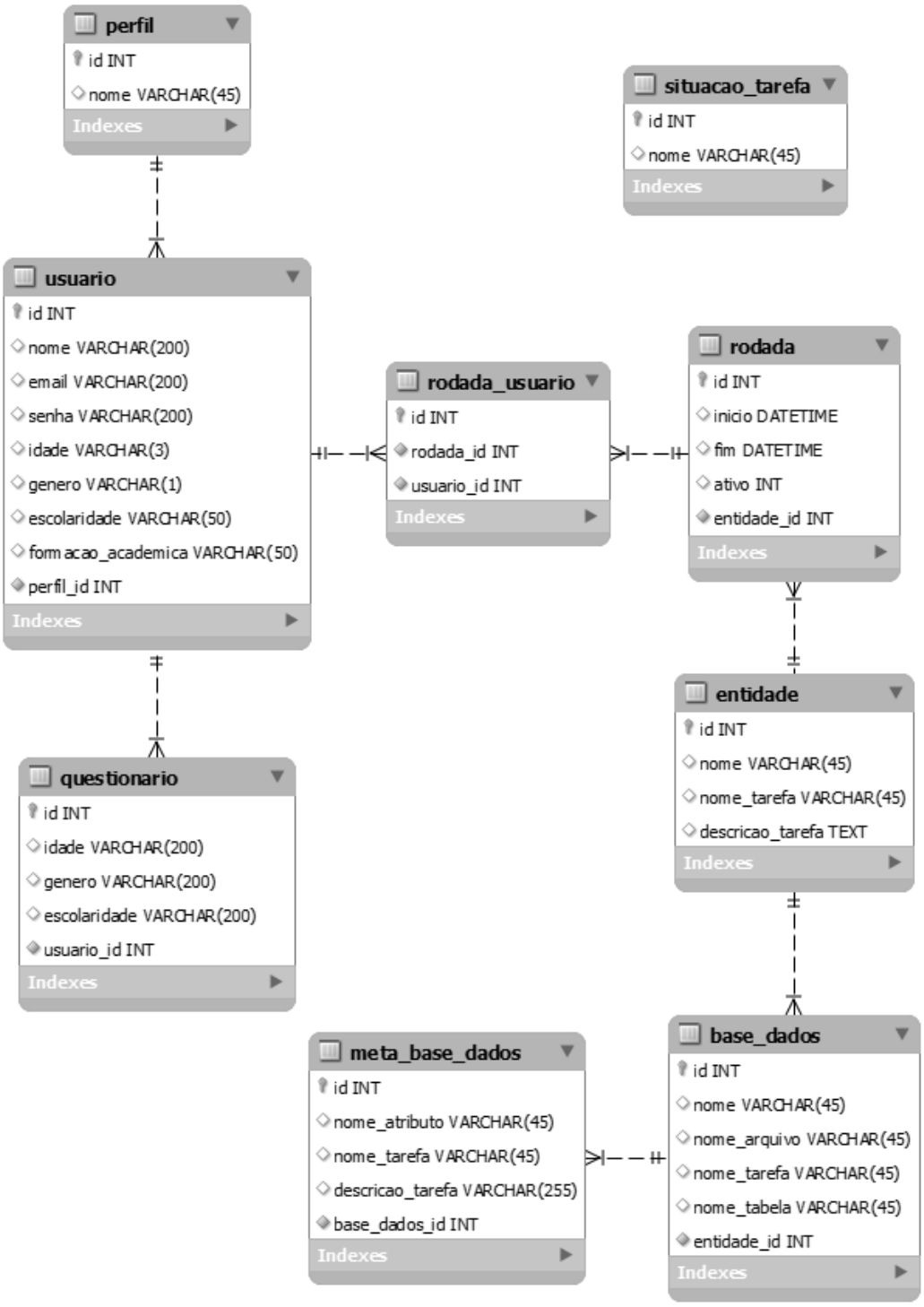


Figura 36 – Modelo de dados da ferramenta CERM (elaborado pelo autor)

Apêndice 3 - Modelo de Dados criado dinamicamente pela ferramenta CERM

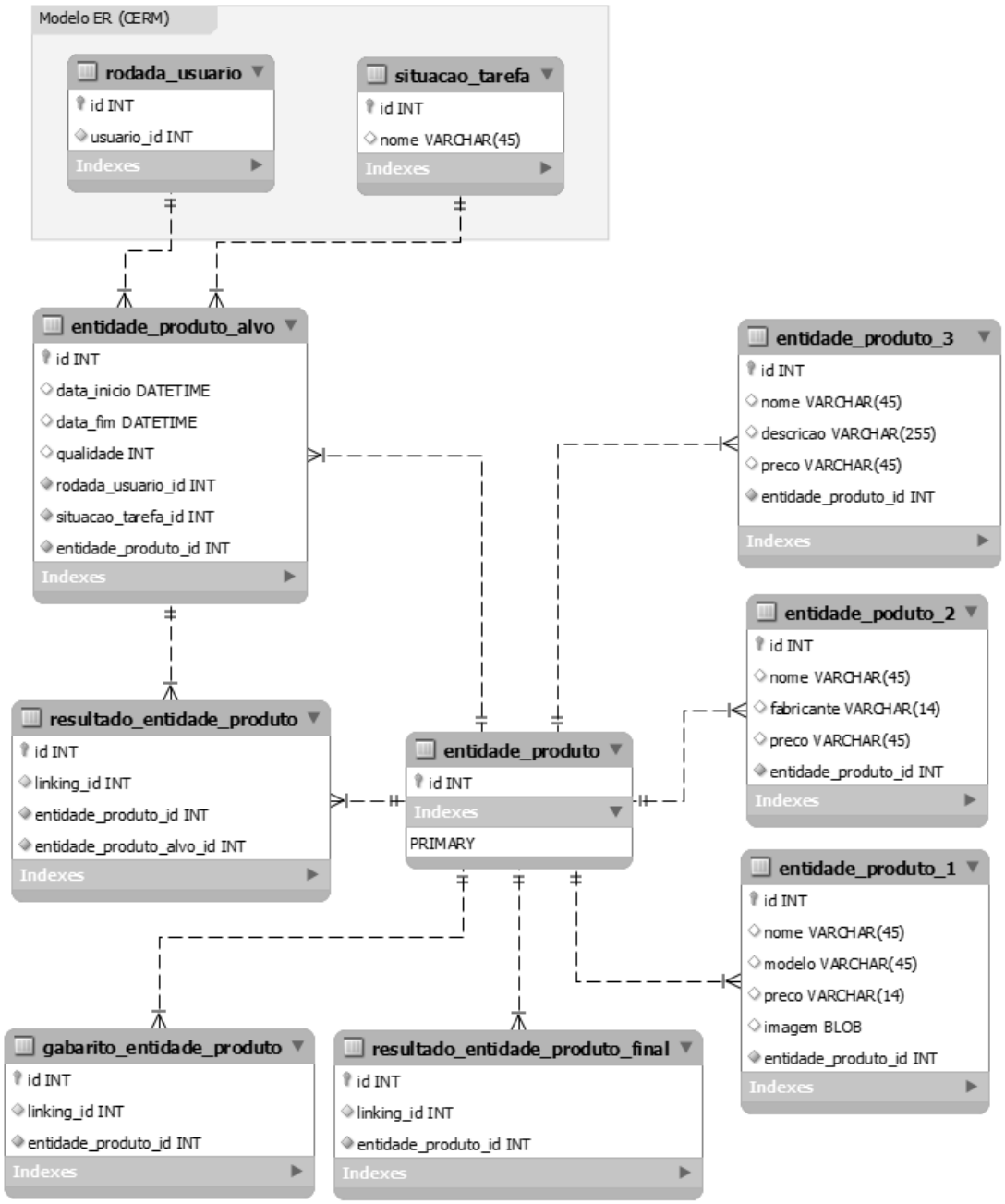


Figura 37 – Modelo de dados criado dinamicamente pela ferramenta CERM ao cadastrar uma entidade (elaborado pelo autor)

Apêndice 4 - Diagrama de Classe da ferramenta CERM

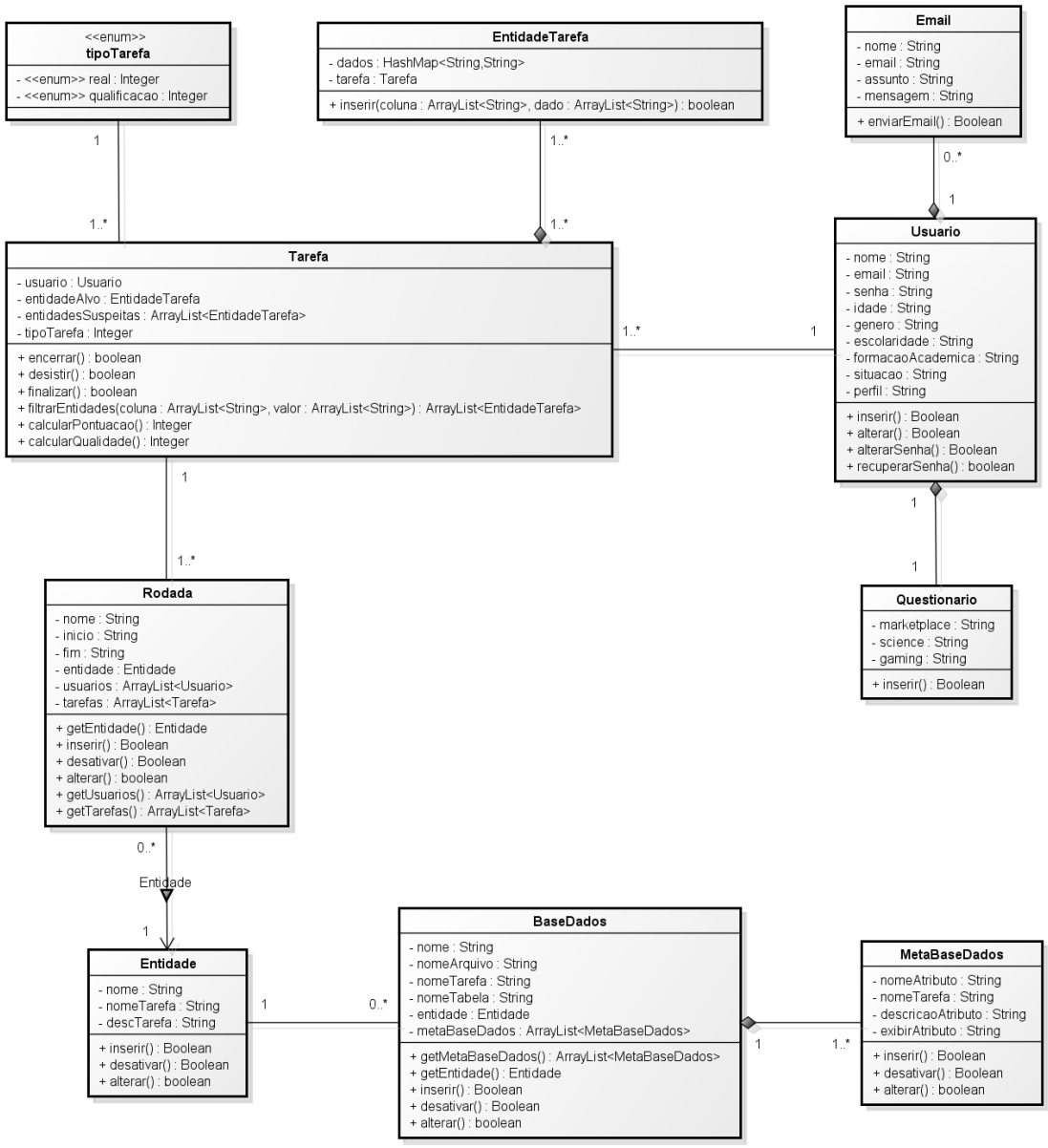


Figura 38 – Diagrama de Classe da ferramenta CERM (elaborado pelo autor)

Apêndice 5 - Diagrama de Caso de Uso da ferramenta CERM

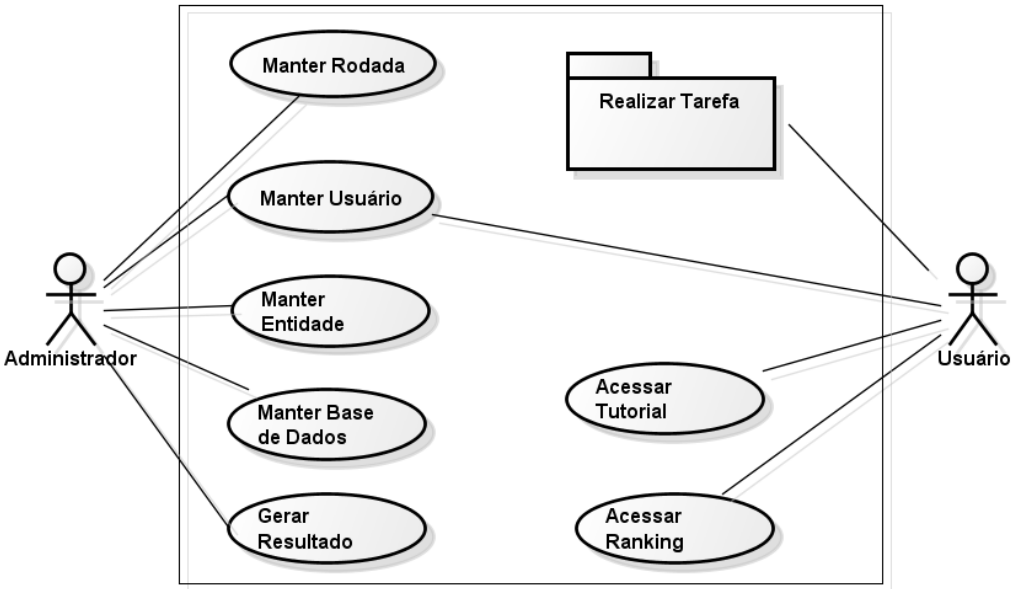


Figura 39 – Diagrama de Caso de Uso da ferramenta CERM (elaborado pelo autor)

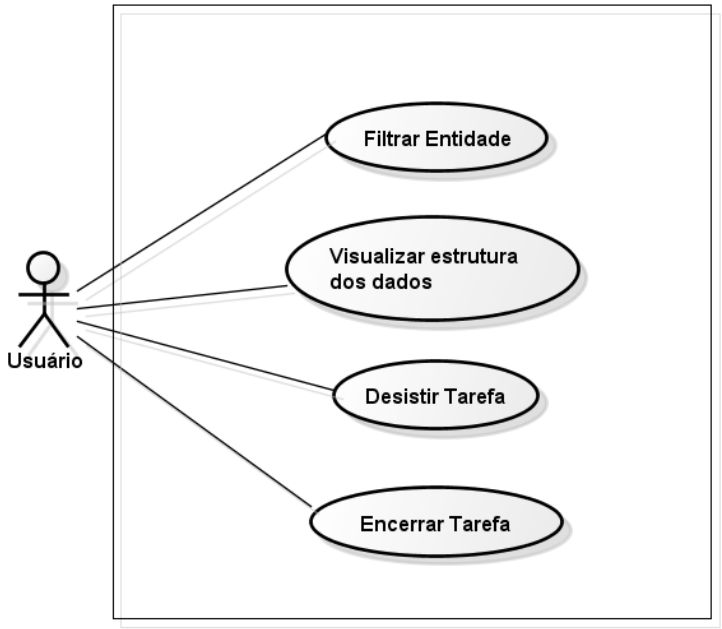


Figura 40 – Diagrama de Caso de Uso Pacote Realizar tarefa da ferramenta CERM (elaborado pelo autor)

Apêndice 6 – Algoritmo para calcular gabarito final a partir dos resultados da multidão

Após a execução da tarefa na fase de resolução, o gabarito ainda não é conhecido. Logo, a única forma de calcular o gabarito nessa fase é através dos resultados anteriores dos usuários.

Para avaliar a similaridade de duas referências, foi utilizada a média ponderada (ALT et al., 2007) entre a opinião das pessoas com relação a semelhança das referências e a qualidade do usuário naquele momento.

Considere um conjunto de pessoas P, um conjunto de qualidades Q e um conjunto de entidades E. Considere ainda duas referências a, b ∈ E, e uma pessoa p ∈ P. A função de *match* para as entidades a e b é dada por:

$$M_p(a,b) = \begin{cases} 1, & se\ a\ pessoa\ relacionou\ as\ referências\ a\ e\ b \\ 0, & caso\ contrário \end{cases}$$

Figura 41 – Função que retorna se usuário agrupou determinada referência com outra (elaborado pelo autor)

A função utilizada para calcular se as referências a e b são a mesma entidade é dada pela fórmula:

$$S(a,b) = \frac{\sum_{p \in P} M_p(a,b) * Q_p}{\sum_{p \in P} Q_p}$$

Figura 42 – Cálculo do grau de similaridade entre duas referências na fase de resolução (elaborado pelo autor)

Cada linha da Tabela 20 indica a situação de um usuário em uma rodada. Para o usuário p1, por exemplo, no momento em que a rodada terminou, ele possuía qualidade 0.9, recebeu como alvo a referência A e a relacionou juntamente com as referências B, C e D.

Tabela 20 – Exemplo de agrupamento de referências equivalentes e sua qualidade (elaborado pelo autor)

Usuário	Qualidade	Entidade Alvo	Suspeita 1	Suspeita 2	Suspeita 3
p1	0,9	A	B	C	D
p2	0,8	A	B	C	
p3	0,4	A	E		
p4	0,2	A	C	D	

Aplicando a função para as referências A e B baseada na Tabela 20, temos que:

$$S(A,B) = \frac{0.9 * 1 + 0.8 * 1 + 0.4 * 0 + 0.2 * 0}{0.9 + 0.8 + 0.4 + 0.2} = \frac{1.7}{2.3} \cong 0.74$$

Figura 43 – Exemplo de aplicação da fórmula para calcular grau de similaridade pela votação da maioria (elaborado pelo autor)

Logo à similaridade das referências A e B de acordo com a opinião da multidão é de aproximadamente 0.74.

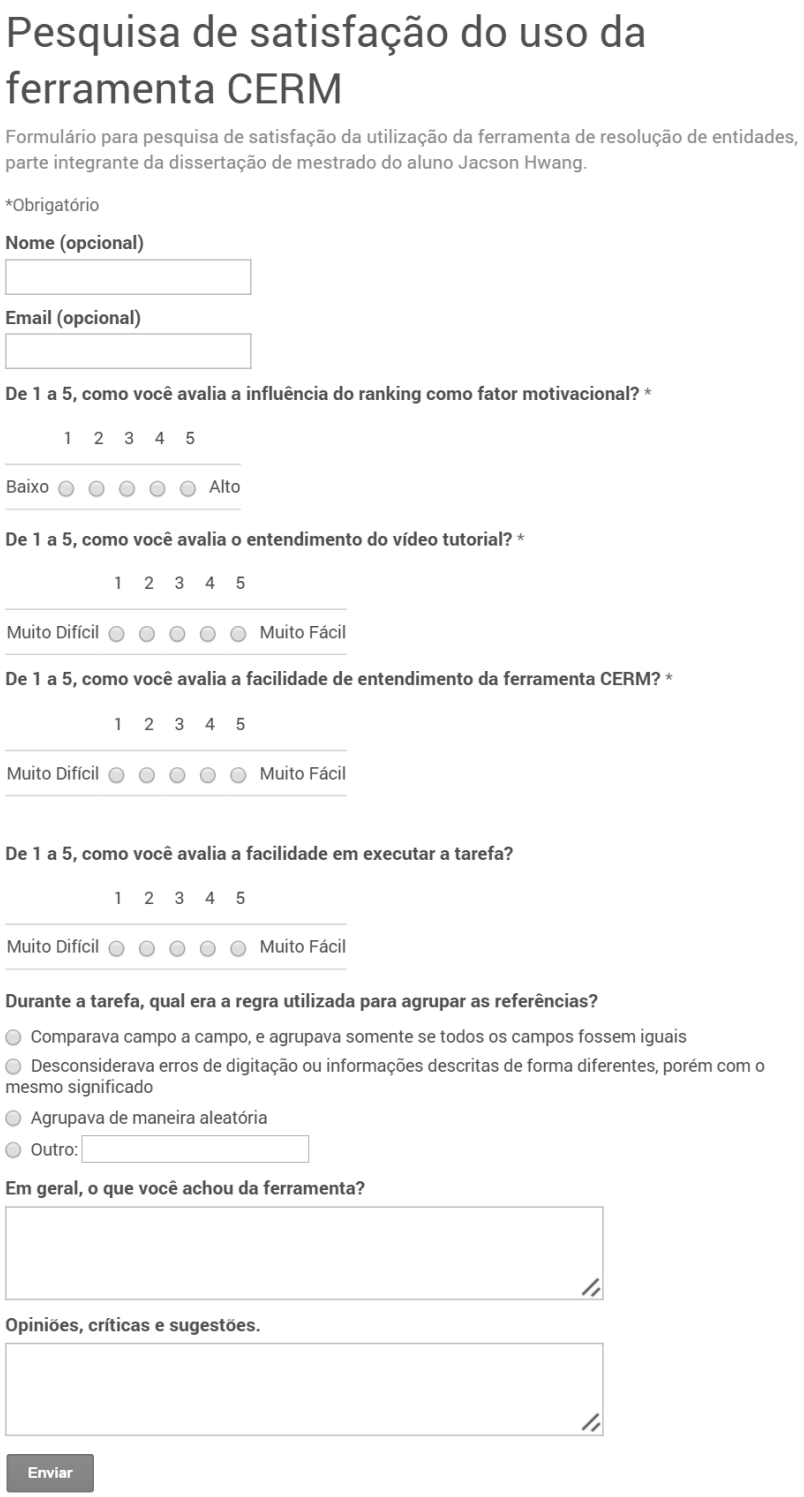
Apêndice 7 - Perfil dos usuários da ferramenta CERM

A Tabela 21 ilustra a resposta de cada usuário cadastrado na ferramenta CERM. A ordem foi determinada pela data e hora de cadastro, e as respostas numéricas equivalem a: (0) Nenhum, (1) Básico, (2) Intermediário e (3) Avançado.

Tabela 21 - Perfil dos usuários da ferramenta CERM (elaborado pelo autor)

Usuário	Nível de Escolaridade	Formação em TI	Conhecimento em Ferramentas Crowdsourcing		Outras experiências
			Marketplace	CrowdScience	Banco de Dados
1	Superior Incompleto	Sim	0	0	1
2	Superior Incompleto	Sim	0	0	2
3	Mestrado	Sim	1	1	3
4	Superior Incompleto	Sim	0	0	1
5	Superior Incompleto	Sim	0	0	1
6	Ensino Médio Completo	Não	0	1	0
7	Superior Completo	Não	0	0	0
8	Superior Incompleto	Sim	0	0	2
9	Mestrado	Sim	0	0	3
10	Superior Incompleto	Não	0	0	0
11	Superior Incompleto	Não	0	0	0
12	Mestrado	Sim	1	1	3
13	Mestrado	Sim	1	1	2
14	Doutorado	Sim	0	0	1
15	Mestrado	Sim	1	2	3
16	Pós-graduação	Não	2	1	0
17	Ensino Médio Completo	Não	0	0	0
18	Mestrado	Sim	0	0	2
19	Superior Incompleto	Sim	0	0	1
20	Superior Incompleto	Sim	1	1	1
21	Pós-graduação	Não	0	0	0
22	Ensino Fundamental	Não	0	0	0
23	Superior Completo	Não	0	0	0
24	Doutorado	Sim	2	2	3
25	Superior Incompleto	Sim	0	0	1
26	Superior Incompleto	Sim	1	1	2
27	Superior Incompleto	Sim	0	0	1
28	Mestrado	Sim	0	0	1
29	Superior Incompleto	Sim	0	1	1
30	Superior Incompleto	Sim	1	2	2
31	Doutorado	Sim	0	0	3
32	Superior Incompleto	Não	0	0	0

Apêndice 8 - Formulário de pesquisa de satisfação da ferramenta CERM



Pesquisa de satisfação do uso da ferramenta CERM

Formulário para pesquisa de satisfação da utilização da ferramenta de resolução de entidades, parte integrante da dissertação de mestrado do aluno Jacson Hwang.

*Obrigatório

Nome (opcional)

Email (opcional)

De 1 a 5, como você avalia a influência do ranking como fator motivacional? *

1 2 3 4 5

Baixo ○ ○ ○ ○ ○ Alto

De 1 a 5, como você avalia o entendimento do vídeo tutorial? *

1 2 3 4 5

Muito Difícil ○ ○ ○ ○ ○ Muito Fácil

De 1 a 5, como você avalia a facilidade de entendimento da ferramenta CERM? *

1 2 3 4 5

Muito Difícil ○ ○ ○ ○ ○ Muito Fácil

De 1 a 5, como você avalia a facilidade em executar a tarefa?

1 2 3 4 5

Muito Difícil ○ ○ ○ ○ ○ Muito Fácil

Durante a tarefa, qual era a regra utilizada para agrupar as referências?

○ Comparava campo a campo, e agrupava somente se todos os campos fossem iguais

○ Desconsiderava erros de digitação ou informações descritas de forma diferentes, porém com o mesmo significado

○ Agrupava de maneira aleatória

○ Outro:

Em geral, o que você achou da ferramenta?

Opiniões, críticas e sugestões.

Enviar

Figura 44 - Formulário de satisfação da ferramenta CERM (elaborado pelo autor)

Apêndice 9 - Resposta dos participantes ao formulário do Apêndice 8

A seguir são apresentadas as tabelas contendo as respostas do formulário de satisfação dos usuários da ferramenta CERM (Figura 44). Das 42 pessoas cadastradas, apenas 10 pessoas responderam o formulário.

As questões e respostas serão referenciadas nas tabelas conforme abaixo:

- **Questão 1**: De 1 a 5, como você avalia a influência do ranking como fator motivacional?
- **Questão 2**: De 1 a 5, como você avalia o entendimento do vídeo tutorial?
- **Questão 3**: De 1 a 5, como você avalia a facilidade de entendimento da ferramenta CERM?
- **Questão 4**: De 1 a 5, como você avalia a facilidade de executar a tarefa?
- **Questão 5**: Durante a tarefa, qual era a regra utilizada para agrupar as referências?
    - **Resposta 1**: Comparava campo a campo, e agrupava somente se todos os campos fossem iguais
    - **Resposta 2**: Desconsiderava erros de digitação ou informações descritas de forma diferentes, porém com o mesmo significado
    - **Resposta 3**: Agrupava de maneira aleatória

Tabela 22 - Resposta para a pesquisa de satisfação (elaborado pelo autor)

Usuário	1	2	3	4	5	6	7	8	9	10	Média
Questão 1	4	5	4	3	4	5	4	2	5	3	3,9
Questão 2	4	3	2	4	4	4	4	4	4	3	3,6
Questão 3	5	4	2	4	4	4	4	3	4	2	3,6
Questão 4	5	4	4	4	3	5	4	3	4	3	3,9
Questão 5	2	1	2	2	1	1	2	2	2	2	-