Harnessing Collective Intelligence on Social Networks

Jon Chamberlain

A thesis submitted for the degree of Doctor of Philosophy (PhD)

School of Computer Science and Electronic Engineering
University of Essex

August 2015

Acknowledgements

My decision to embark on the PhD journey was sealed several years prior by Dr Doug Arnold who, during a chance conversation with my eventual supervisor, advised my employment based on my experience in the Web industry. That decision led me to work with Professor Udo Kruschwitz and Professor Massimo Poesio on the development of the *Phrase Detectives* project. They both kindly agreed to jointly supervise my PhD and have provided valued support and insight. My third supervisor, Professor Dave Smith, helped keep my research grounded in the context of marine conservation.

One of the most valuable events I attended during my candidacy was the Doctoral Consortium of HCOMP14. The students and academics I met had a profound effect on my ability to communicate my research ideas. Closer to home, my colleagues at the University of Essex, in particular in the Language and Computation (LAC) research group, have provided much needed guidance and light relief from the pressures of this undertaking.

I appreciated the opportunity to discuss my work with the examiners of this thesis: Professor Johan Bos from the University of Groningen, whose interests in computational linguistics and natural history reflect my own; and Professor Richard Bartle, whose attention to detail and knowledge of game theory is unsurpassed.

Finally, my thanks go to Sally, Dylan and the rest of my family for supporting me through this journey.

This research was partially funded by the EPSRC Doctoral Training Allowance granted by the University of Essex.

Abstract

Crowdsourcing is an approach to replace the work traditionally done by a single person with the collective action of a group of people via the Internet. It has established itself in the mainstream of research methodology in recent years using a variety of approaches to engage humans in solving problems that computers, as yet, cannot solve.

Several common approaches to crowdsourcing have been successful, including peer production (in which the participants are inherently interested in contributing), microworking (in which participants are paid small amounts of money per task) and games or gamification (in which the participants are entertained as they complete the tasks).

An alternative approach to crowdsourcing using social networks is proposed here. Social networks offer access to large user communities through integrated software applications and, as they mature, are utilised in different ways, with decentralised and unevenly-distributed organisation of content.

This research investigates whether collective intelligence systems are facilitated better on social networks and how the contributed human effort can be optimised. These questions are investigated using two case studies of problem solving: anaphoric coreference in text documents and classifying images in the marine biology domain.

Social networks themselves can be considered inherent, self-organised problem solving systems, an approach defined here as groupsourcing, sharing common features with other crowdsourcing approaches; however, the benefits are tempered with the many challenges this approach presents. In comparison to other methods of crowdsourcing, harnessing collective intelligence on social networks offers a high-accuracy, data-driven and low-cost approach.

Contents

List of Figures

LIST OF FIGURES

List of Tables

1

Introduction

1.1 Motivation

In 2008 a meeting of coral reef specialists from around the world estimated that 19% of the world's reefs were effectively lost, with a further 35% percent seriously threatened in the next 20 to 40 years. By 2050 all coral reefs are estimated to be at risk from human activities including tourism, coral mining, pollution, overfishing, canal dredging and the warming and acidification of oceans [Hoegh-Guldberg et al., 2007; Wilkinson, 2008]. This rapid decline will have a catastrophic impact around the world with the total net value of the world's coral reef ecosystems estimated to be close to $30 billion per year [Cesar and Pet-Soede, 2003] and two-thirds of humans living within 100 kilometres from the ocean [Burke et al., 2001].

Within one generation our world will have irreversibly changed for the worse and given the current global priorities of economic growth, energy security, threats of terrorism, and pandemics[1] there is not likely to be a change in policy or increase in funding for conservation, monitoring or research. It is apparent that we need a radical solution for monitoring marine biodiversity that can collect vast amounts of data and process it for actionable knowledge.

One solution is born from one of the threats itself: the explosion of recreational SCUBA diving. Coupled with the affordability of underwater digital camera equipment,

Portions of this chapter previously appeared in Chamberlain, Kruschwitz, and Poesio [2012]; Poesio et al. [2013].

[1] `https://g20.org/wp-content/uploads/2014/12/brisbane_g20_leaders_summit_communique.pdf`

Figure 1.1: The start of the script for Monty Python's Dead Parrot (The Pet Shoppe) sketch, highlighting ambiguity. What entity does *Miss* refer to? From the context of the script we would assume it to be *The owner*; however, in the scene it is Michael Palin playing the part of a male shopkeeper, therefore the feminine title of *Miss* shouldn't be applied. This is an example of a linguistic referencing problem that a human can easily solve (and find funny) but a computer would find difficult because it logically doesn't make sense.

more data are being created and shared in informal ways, such as on social networks[1], with data being annotated by an enthusiastic community on a scale never been seen before. With more marine ecosystems being monitored by the public in this way a huge resource is being created and this research lays the foundations for developing a full-scale solution to the problem of monitoring the health of the world's oceans with the collective intelligence of social networks (see Section 7.11 for progress towards developing a prototype application).

In response to these large-scale challenges, this research investigates harnessing collective intelligence on social networks and aims to utilise techniques of text analytics, crowdsourcing and social network analysis to understand better how the data can be processed. The ultimate goal is to demonstrate that social networks are inherent problem-solving platforms that are comparable, if not superior, to existing approaches to creating and curating large knowledge resources. This hypothesis is tested on the common problems of **understanding human language and classifying images**.

1.2 Synopsis

Ever since the shift towards statistical methods, research in human language technology has been driven by the availability of large-scale resources (corpora, lexica and, more recently, repositories of encyclopedic knowledge). The creation of such resources

[1]Social networks in this context refer to software applications that allow Internet users to share information, further defined in Section 2.4.4.

Figure 1.2: An image of a school of Red Sea Bannerfish, highlighting some of the difficulties image classifiers have when identifying and counting the objects (in this case fish) in the image, such as partial objects, occlusion, rotation, contrast and depth of field.

has traditionally been the task of dedicated experts who did their work manually. Extracting information from structured document collections (e.g. databases and text with predictable layout) is relatively straight-forward. However, the vast majority of documents consist of unstructured natural human language (including the Internet) and processing such big data sources is on a scale traditional manual methods are not designed for. Furthermore these types of documents may contain more examples of ambiguity that make them harder for machine to understand (see Figure 1.1 for an example).

Interpreting and classifying images has also been an active area of research, and large-scale resources are required to train and test systems that attempt to do the task automatically (see Figure 1.2 for an example). The sharing of multimedia content has become widespread to a scale at which traditional methods of classification are no longer adequate.

The first obstacle is how to **overcome the bottleneck in collecting and annotating data**. Collecting the primary data to answer research questions is a resource-

intensive and time-consuming process in which traditionally the data annotation would be done by a handful of paid annotators who are trained in the specific annotation task required from the data. Their efforts would perhaps be validated by other experts and inconsistencies would be resolved. This would produce a data set called a **gold standard** that could be considered a set of correct answers or labels to the primary data [Poesio and Artstein, 2008].

This methodology does not capture ambiguities in the data and these are often the cases that are most interesting. **Unusual or ambiguous data** will cause annotators to mark up the data in different ways and by preserving this conflict of ideas it would be possible to highlight the most interesting problems when automatically processing the data. These cases present the same challenge in text and image data, and in different domains. The problem is more acute when you consider the different levels of experience and training the annotators may have.

Collective intelligence can be shown in many domains including Computer Science, Economics and Biology[1], but here we focus on coordinating collective action in computational systems. Individual decisions made by a community of users (or annotators) are aggregated in an attempt to produce a high-quality, collective decision comparable to an expert judgement [Surowiecki, 2005]. This is motivated by the observation that a group of individuals can contribute to a collective solution, which has a better performance and is more robust than an individual's solution, for example, in simulations of collective behaviours in self-organising systems [Johnson et al., 1998].

Crowdsourcing is an approach to replace the work traditionally done by a single person by the collective action of a group of people via the Internet [Howe, 2008]. Crowdsourcing has established itself in the mainstream of research methodology in recent years using a variety of approaches to engage humans to solve problems that computers, as yet, cannot solve. Whilst the concept of **human computation** [von Ahn, 2006] goes some way towards solving problems, it also introduces new challenges for researchers, not least how to deal with human psychology.

Several common approaches to crowdsourcing have been successful. In the first approach, **peer production**, the user is inherently interested in contributing, for example Wikipedia. In a second approach, **microworking**, participants are paid small amounts

[1] `http://scripts.mit.edu/$\sim$cci/HCI`

of money per task, for example Amazon Mechanical Turk.[1] A third approach is to entertain the user whilst they complete tasks, typically using games or gamification. This **game-with-a-purpose (GWAP)** approach has been used for many different types of crowdsourced data collection including text, image, video and audio annotation, biomedical applications, transcription, search and social bookmarking [Chamberlain et al., 2013].

These crowdsourcing methods are typically focused on getting users to complete tasks preset by an administrator or organisation (called a 'requester' in microworking); however, the problem-solving abilities of a crowd can also been seen in **Community Question Answering (cQA)** websites in which an active online community present and resolve problems without a central administrative structure.

Social networks such as Facebook[2], LinkedIn[3] and Flickr[4] offer access to large user communities through integrated software applications. As social networks mature the software is utilised in different ways, with decentralised and unevenly-distributed organisation of content, similar to how Wikipedia users create pages of dictionary content or questions are posed on cQAs. **Citizen science**, in which members of the public contribute knowledge to scientific endeavours, is an established predecessor of crowdsourcing and social networks have been successfully used to connect professional scientists with amateur enthusiasts [Gonella, Rivadavia, and Fleischmann, 2015; Sidlauskas et al., 2011].

This research investigates whether collective intelligence systems are better facilitated on social networks, whether the contributed human effort can be optimised and whether social networks themselves can be considered inherent, self-organised problem-solving systems. These questions are investigated using two case studies of problem solving: anaphoric coreference in text documents and image classification in the marine biology domain.

[1] `https://www.mturk.com`
[2] `https://www.facebook.com`
[3] `https://www.linkedin.com`
[4] `http://www.flickr.com`

1.3 Research questions

The primary research question is whether collective intelligence on social networks can be used to create large-scale **data** resources, with high-quality labelling of **information** about the data, that can be used to create **knowledge** to solve problems that cannot be addressed in any other way. This question makes one important assumption: that social networks can be viewed as problem-solving systems. If this assumption holds true, then a wealth of ideas and research regarding crowdsourcing and collective intelligence analysis is at our disposal. This assumption is investigated in Chapter 3.

1: Can a problem-solving system deployed on a social network gather more answers of a higher quality than a standalone system? Social networks have large numbers of users so it is intuitive to believe that a system deployed on them would benefit from increased exposure to a larger user base and therefore participation would increase, especially if the system was integrated into the social features. Additionally, social networks work hard to ensure their users are real people and not companies, groups or spam [Stringhini, Kruegel, and Vigna, 2010] so the chance of poor-quality answers being submitted might be lower. These issues are investigated in Chapter 4 using *Phrase Detectives*, an online game designed to collect annotations about human language, with one system deployed as a standalone game and another deployed on the social network Facebook.

2: Can the standard annotation model be improved upon to make the most of the efforts of human annotators? It is a well-studied phenomenon that a group of non-experts can perform as well as, if not better than, a single expert at problem solving (see Chapter 2); however, can a more sophisticated model be used in which the collected decisions are also validated by the users?

This raises the question of whether gathering more opinions would be as valuable as validating existing opinions, therefore optimisation of the model is also considered.

Additionally, the question of answer confidence is raised, in particular in problems where there may be more than one correct solution (or no best solution). These issues are investigated using a model proposed in Section 3.2.3 and evaluated in the *Phrase Detectives* game in Chapter 5.

3: Is problem solving inherent on social networks and, if so, can the data be analysed using the same techniques developed for crowdsourcing? The final question explores the idea that problem solving is an inherent part of the way humans interact with each other on social networks and that it can be viewed in the same way as a crowdsourcing system. The methods and techniques investigated in Chapter 4 and 5 are applied to social network groups in which users solve image classification tasks (see Chapter 6).

The benefits of using a crowd to help solve data-annotation problems are tempered by the many challenges these approaches present. As well as having to deal with human psychological and sociological issues, there are issues of ethics and workers' rights. Although humans are used for computation, they can not be treated as one treats computers and resources cannot be acquired in the same way. These issues are discussed in more depth in Chapter 7.

1.4 Contribution

This research offers several contributions:

- A detailed overview of crowd-based approaches to text and image annotation, comparing factors such as cost, speed, and quality. These approaches are also compared by their features to discover similarities that allow them to be discussed with a common terminology;

- A definition of social networks as problem-solving platforms in the same terms as other crowd approaches using the same terminology and features. This thesis even goes as far as defining a new term 'groupsourcing' in order to clarify the difference between using social networking groups and other crowdsourcing approaches;

- Analysis of the benefits of deploying a crowdsourcing system on social networks, which shows there are numerous benefits and limitations that should be considered;

- A detailed analysis of how validation can be used to improve on the performance of a standard annotation model;

- Analysis of inherent social network problem solving showing very high (near-expert) accuracy on difficult image classification tasks;
- A prototype system for viewing the aggregated knowledge of social networks;
- The development of openly-accessible tools for researchers to investigate these ideas further, as well as the final analysed datasets that allow researchers access to large, collaboratively-created resources.

1.5 Published work

Some work has been published in papers in which the primary contributor was the author of this thesis and each chapter begins with a declarative footnote. These include:

- A full description of the *Phrase Detectives* system [Poesio et al., 2013], which incorporated a number of previous papers [Chamberlain, Kruschwitz, and Poesio, 2009; Chamberlain, Poesio, and Kruschwitz, 2009; Chamberlain, Poesio, and Kruschwitz, 2008];
- Analysis of user performance data from *Phrase Detectives* [Chamberlain and O'Reilly, 2014];
- Definition and simulation of the Annotation Validation (AV) Model [Chamberlain, 2014a];
- Discussions of using a gaming approach to collecting data [Chamberlain et al., 2013; Chamberlain, Kruschwitz, and Poesio, 2013];
- An initial investigation into deploying games on social networks [Chamberlain, Kruschwitz, and Poesio, 2012];
- Definition and initial analysis of the groupsourcing approach, along with details of a prototype system [Chamberlain, 2014b,c];

PART I: Approaches to Collaborative Annotation

2

Related work

Related work of this research focuses primarily on approaches to harnessing collective intelligence from a group of people in order to solve a particular problem or task. This can be done by developing structured systems for collecting data (a common approach to crowdsourcing) or by data mining and information extraction. Once the data have been acquired from the crowd it must be processed or aggregated in some way to produce a set of answers to the task.

The primary research question is whether collective intelligence on social networks can be used to create large-scale **data** resources, with high-quality labelling of **information** about the data, that can be used to create **knowledge** to solve problems that cannot be addressed in any other way. There are three ways information can be added to data. It can be added at the point of creation, most usually by the person who created the data, but also by the device that was used. For example, a camera will record EXIF information with every image taken which includes information about the manufacturer of the camera, the lens settings, GPS coordinates, etc.

Information can also be added by processing. This step takes the data and applies algorithms that try to understand the data. Depending on the data type, preprocessing can be very accurate, but is more normally error-prone and needs supervision from administrators.

Finally, information can be added manually after the data have been created and

Portions of this chapter previously appeared in Chamberlain and O'Reilly [2014]; Chamberlain et al. [2013]; Chamberlain, Kruschwitz, and Poesio [2012, 2013]; Chamberlain [2014a,b,c]; Poesio et al. [2013].

this is done using an annotation task. The annotation task can take many forms and levels of complexity depending what will be annotated and who will do the task. It is this latter case that is of most interest here in the areas of natural language processing and image classification.

2.1 Natural language processing

The first annotated corpora, such as the one million word Brown Corpus [Kucera and Francis, 1967], were only concerned with low-level linguistic information such as lemmas and part-of-speech tags, and were created entirely by hand. This methodology is still used for the majority of annotation projects, in particular for projects concerned with the annotation of more complex types of linguistic information, and arguably still has a place to create resources of very high quality but the costs involved are considerable. Thanks to substantial investments in Germany and the USA, such as the funding of SALSA [Burchardt et al., 2009] and OntoNotes [Hovy et al., 2006; Pradhan et al., 2007], it has been possible to create Brown Corpus-size annotated corpora for semantic tasks such as coreference, predicate argument structure and word sense disambiguation. However, the costs required (in the order of over one million dollars per million words of annotated data for each level) make it clear that the traditional hand-annotation methods used in such projects are not feasible to annotate larger amounts of data.

A partly-validated type of annotation also involves the development of a formal coding scheme and training of annotators, but most items will be typically annotated only once, for example, in the ARRAU [Poesio and Artstein, 2008] and GNOME [Poesio, 2004a] corpora for anaphoric co-reference.

A faster and cheaper semi-automatic methodology has therefore become standard to annotate larger amounts of linguistic information for which relatively high-quality annotation systems existed. When this is the case, a preliminary annotation with automatic methods is followed by partial hand-correction. The methodology was pioneered in the annotation of the British National Corpus (BNC), the first 100 million word linguistically-annotated corpus [Burnard, 2000], thanks to the availability of relatively high-quality automatic part-of-speech taggers trained on smaller scale data. With the development of the first high-quality chunkers this methodology became applicable to the case of syntactic annotation as well, and was used for the creation of the Penn

Treebank, although more substantial hand-checking was required [Marcus, Santorini, and Marcinkiewicz, 1993].

Semi-supervised and unsupervised processes using statistical and machine learning techniques do not require much human intervention and the rules are learnt automatically. These techniques started with decision trees and Hidden Markov Models [Klein et al., 2003] and have advanced to more promising techniques including Maximum Entropy Markov Models [McCallum, Freitag, and Pereira, 2000] and Conditional Random Fields [Banko and Etzioni, 2008; Culotta, McCallum, and Betz, 2006]. These have a lower accuracy compared to supervised processes; however, systems such as *TextRunner* [Banko et al., 2007] can be applied to any domain and work with very large document collections.

In a more recent approach, called active annotation, the activity of annotation is guided by the needs of the system being trained [Settles, 2009; Vlachos, 2006].

Weakly-supervised techniques have proven effective for tasks such as named entity resolution, word sense disambiguation, and relation extraction, in which collaboratively created resources such as Wikipedia can be used to generate the training data [Mintz et al., 2009]. No such resources are available for a number of core human language tasks, including coreference, predicate argument structure, and discourse structure; however, recent projects such as the Groningen Meaning Bank [Basile et al., 2012] use a variety of methods to create a large semantically-annotated corpus.

Anaphoric coreference Anaphora resolution is a key semantic task both from a linguistic perspective and for applications ranging from summarisation to text mining, but one for which medium-sized corpora have only recently become available and our understanding of which is not such that linguists can produce a coding scheme with high reliability [Poesio and Vieira, 1998; Zaenen, 2006].

(2.1) Wivenhoe developed as a port and until the late 19th century was effectively a port for Colchester, as large ships were unable to navigate any further up the River Colne, and had two prosperous shipyards. It became an important port for trade for Colchester and developed shipbuilding, commerce and fishing industries. The period of greatest prosperity for the town came with the arrival of the railway in 1863.[1]

[1] `http://en.wikipedia.org/wiki/Wivenhoe`

Anaphora is the linguistic mechanism of referring back to an entity already introduced in a discourse, e.g. *Wivenhoe* in Example 2.1, sometimes using the same expression again (as in the case of the two references to *Colchester* in the same example), but in many other cases using different expressions (as in the two other references to Wivenhoe in the example using *it* and *the town*).

Interpreting anaphoric coreference therefore involves, first of all, keeping track of which entities have been mentioned by building a discourse model [Kamp and Reyle, 1993]. Whenever a new linguistic expression of interest is encountered (such expressions are usually called **markables** in an annotation context) the reader or system has to decide whether this markable introduces a new entity (in which case it is called **discourse-new** [Prince, 1992]) or whether instead it refers to an entity already introduced and if so, which one. This entity is called the **antecedent** and the term **discourse-old** is used to indicate expressions which refer to a previously introduced antecedent. For example, in the second sentence in Example 2.1, the pronoun *it* could refer to *Wivenhoe*, *Colchester*, or indeed *the River Colne*; whereas in the third sentence, the markable *the town* could be interpreted as having either *Wivenhoe* or *Colchester* as the antecedent.

The problem of interpreting such markables is further complicated by the fact that not all nominal phrases in English are referential, i.e. either introduce a new entity or refer to one already introduced. Expressions such as 'it' or 'there' may have no semantic content at all. For example, in Example 2.2, *It* is only used for syntactic reasons and is semantically empty. Many nominal phrases are also used to express **properties** of entities, as opposed to referring to entities directly. For example, the markable *a fireman* in Example 2.3 is used to express a property of the entity referred to by the subject of the sentence, *Sam*.

(2.2) It is raining.

(2.3) Sam is a fireman.

Choosing the logical form content of a noun phrase (referring, empty, property) or an antecedent between the entities already introduced in discourse may not be easy tasks, and in many cases the text does not provide enough information to decide.

Consider the instance of *it* in utterance 5.1 in Example 2.4. In experiments, subjects were asked about the interpretation of this markable and two thirds of the subjects chose *engine E2*, whereas the other third chose *the boxcar at Elmira* [Poesio et al., 2006].

(2.4)
```
3.1    M: can we .. kindly hook up
3.2    : uh
3.3    : engine E2 to the boxcar at .. Elmira
4.1   S: ok
5.1   M: +and+ send it to Corning
5.2    : as soon as possible please
```

These difficulties in interpretation suggest the need to collect multiple judgements for each expression and in cases of disagreement it may be best to preserve such judgements rather than attempting to make a choice between them, i.e. create a set of answers rather than the best answer.

2.2 Image classification

Categorising and classifying images, as well as the entities contained within them, has been the long-term goal for computer vision (Barnard et al., 2003); however, only in the last few decades have screen-based images and digital photography made image classification so ubiquitous, and so important. Machine-readable images have application in robotics, augmented reality, surveillance, face recognition and many other automated tasks that require the mass consumption of imagery on a scale not possible for human administrators to keep up with.

It is therefore not surprising that automatic image annotation is an active area of research [Lu and Weng, 2007] with specific industry-supported tracks, such as Yahoo's Flickr-tag challenge at ACM Multimedia 2013.[1]

Images can have three kinds of annotation applied to them: the entire image can be labelled; regions can be labelled; or specific objects can be outlined and labelled (see Figure 2.1). The objects within the image can then be recognised, for example, by recognition-by-component theory, in which all three-dimensional components can be represented as basic shapes, named *geons*. Research suggests there may be as few

[1] http://acmmm13.org/submissions/call-for-multimedia-grand-challenge-solutions/yahoo-large-scale-flickr-tag-image-classification-challenge

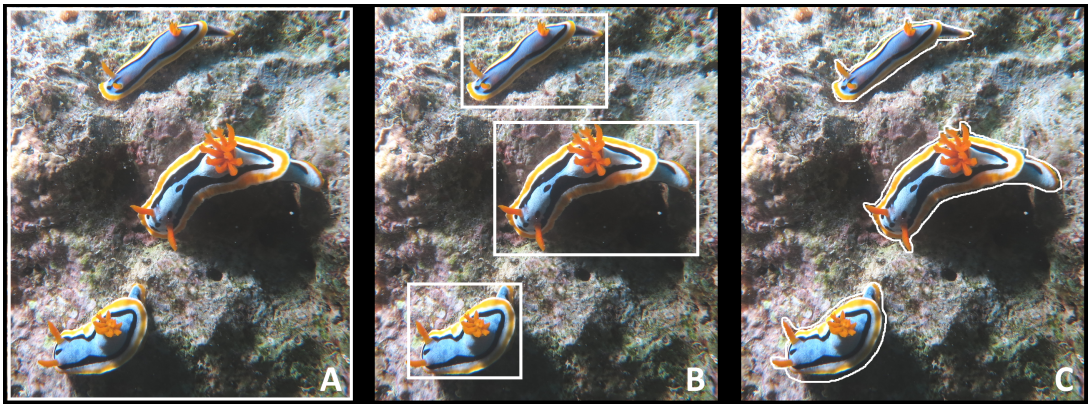


Figure 2.1: Three images showing the different styles of image annotation: entire image labelled (A); regions labelled (B) and object outline labelled (C).

as 36 geons in everyday visual objects making the task of computer vision achievable with enough training materials [Biederman, 1987].

In order to test automatic methods a number of gold standard datasets have been produced, including COIL [Roberts, 1963], Caltech 101 [Fei-Fei, Fergus, and Perona, 2004], and the PASCAL VOC Detection Challenge corpus [Everingham et al., 2010].

Other efforts to create large training resources attempted to align the image classification with the lexical resource of WordNet [Fellbaum, 1998], such as ImageNet which initially contained 3.2M high-resolution images for 5,247 nouns [Deng et al., 2009], although now is considerably larger. Another effort collated 80M images across the entire WordNet noun set (75,062 nouns); however, reported error rates vary between 25-80%, in particular for general concepts [Torralba, Fergus, and Freeman, 2008]. Another effort called BabelNet attempted to merge Wikipedia and WordNet to map the concepts of images associated with Wikipedia pages [Navigli and Ponzetto, 2012].

As well as object identification in images, other research has focused on trying to understand the image scene, such as the Scene UNderstanding (SUN) database that contains 899 categories and 130,519 images [Xiao et al., 2010], as well as the attributes of the image such as 'Is the team in this image winning?' or 'Is this dress fashionable?' [Donahue and Grauman, 2011].

These datasets tend to be biased because the images selected for the corpora have been chosen by criteria, perhaps by subject but also by image quality. Issues such as illumination, pose, clutter, occlusions and viewpoint may all be pre-filtered out and

therefore the training sets may not include the difficult and ambiguous examples.

It could be argued that more fine-grained image analysis is essential to separate the different concepts within an image. To this end, tools have been developed to outline individual elements in an image; however, square and polygon vectors make the image analysis considerably more complex and error-prone. One such effort, *LabelMe*, is an open source database of images and a polygon-drawing tool, with 10,000 images, a third of which have been labelled with complex polygons [Russell et al., 2008].

Identifying marine species in images This research investigates image classification (in which objects in an image are identified) in the domain of marine biology. In this case the annotations are open (can be any text), although they are later normalised to an ontology, and apply to the whole image.

Gold standard image datasets exist for images of wildlife, such as Caltech-UCSD Birds 200, a repository of 200 species of birds displayed in 6,033 images [Welinder et al., 2010a]. More recently, in 2014-15, there have been ImageCLEF challenges focused on automatically identifying several species of fish from video still images.[1]

Analysis of marine species in images has recently become important due to the increasing use of Autonomous Underwater Vehicles (AUV) that can collect data for many hours at a time. These images are either very numerous or very complex in their content, or perhaps both, making it impossible for human annotators to assess the contents of images on a large scale.

It is very apparent how difficult and monotonous the task of annotating deep sea benthic AUV images is even for the most dedicated experts. It is doubtful whether large amounts of images could ever be annotated completely and correctly even by expert annotators so alternative approaches need to be considered. An example of this problem is with deep-sea image annotation to identify habitat assemblage [Bullimore, Foster, and Howell, 2013]. Reanalysis of the data by a single expert showed 47% of assemblages were incorrectly classified [Henry and Roberts, 2014].

For easy-to-identify taxa, both non-expert and automatic systems achieve comparable results to that of experts; however, more difficult groups present problems for all annotation methods. Several notable efforts to classify the habitat shown in an image

[1] `http://www.imageclef.org/2014/lifeclef/fish`

by automatic species recognition have shown some success at the broadest classification levels, such as *iSIS* which achieves 84% overall accuracy [Schoening et al., 2012] and *DiCANN* which achieves 90% accuracy for easy-to-classify images [Culverhouse et al., 2003]. Other efforts have reported higher accuracy of 92-95% with semi-supervised classification [Beijbom et al., 2012].

Inter-annotator accuracy for species classification varies greatly depending on the species being examined (from 35-97%) [Schoening et al., 2012] and intra-annotator consistency is also variable (between 67-87%) [Culverhouse et al., 2003].

It has been suggested that 'obtaining genus or species level data from even the highest quality digital images is very challenging and not without the possibility of human error' and that machine learning will be limited by the gold standards created in this way [Henry and Roberts, 2014].

2.3 Crowdsourcing and collective intelligence

Collective intelligence has been described as 'a form of universally distributed intelligence, constantly enhanced, coordinated in real time and resulting in the effective mobilisation of skills' [Levy, 1997] or perhaps put more concisely: 'where groups of individuals do things collectively that seem intelligent' [Malone, Laubacher, and Dellarocas, 2009]. Collective intelligence can be shown in many domains including Computer Science, Economics and Biology[1], but here we focus on coordinated collective action in computational systems that overcome the bottleneck in creating and curating resources which would normally have been done by experts and/or administrators.

The utility of collective intelligence came to the fore when it was proposed to take a job traditionally performed by a designated employee or agent and outsource it to an undefined large group of Internet users through an open call. This approach, called **crowdsourcing** [Howe, 2008], revolutionised the way traditional tasks could be completed and made new tasks possible that were previously inconceivable due to cost or labour limitations. A survey of 209 documents related to crowdsourcing revealed 40 unique definitions for the term and an authoritative definition has been proposed:

> Crowdsourcing is a type of participative online activity in which an individual, an institution, a non-profit organization, or company proposes to a

[1] `http://scripts.mit.edu/$\sim$cci/HCI`

> group of individuals of varying knowledge, heterogeneity, and number, via a flexible open call, the voluntary undertaking of a task. The undertaking of the task, of variable complexity and modularity, and in which the crowd should participate bringing their work, money, knowledge and/or experience, always entails mutual benefit. The user will receive the satisfaction of a given type of need, be it economic, social recognition, self-esteem, or the development of individual skills, while the crowdsourcer will obtain and utilize to their advantage what the user has brought to the venture, whose form will depend on the type of activity undertaken.
>
> [Estellés-Arolas and González-Ladrón-De-Guevara, 2012]

Whilst crowdsourcing has established itself in the mainstream of research methodology, issues of participant recruitment and incentivisation are significant and many projects do not live up to expectations because human effort cannot be acquired in the same way as machines.

It has been proposed there are four main categories of crowdsourcing [Brabham, 2013]:

1. Knowledge discovery, in which users find and organise information (e.g. SeeClickFix[1]);
2. Broadcast search, in which users solve empirical problems (e.g. InnoCentive[2]);
3. Peer-vetted creative production, in which users create resources the worth of which is judged by the community (e.g. Threadless[3]);
4. Distributed human intelligence tasking, in which users solve tasks of different complexity.

Distributed human intelligence tasking combines collective intelligence, crowdsourcing and human computation to enable a large group of collaborators to work on tasks normally done by highly-skilled (and highly-paid) annotators and aggregates their collective answers to produce a more complex dataset that not only is more robust than

[1] `http://www.seeclickfix.com`
[2] `http://www.innocentive.com`
[3] `https://www.threadless.com`

an individual answer but allows for ambiguity. Enabling groups of people to work on the same task over a period of time is likely to lead to a collectively intelligent decision [Surowiecki, 2005]. This research focuses on this category of crowdsourcing and explores the approaches to engaging a crowd to solve natural language processing and image classification problems.

2.3.1 User motivation and participation

There are three main incentive structures that can be used to motivate users: personal; social; and financial [Chamberlain, Poesio, and Kruschwitz, 2009]. These directly relate to other classifications of motivations in previous research: Love; Glory; and Money [Malone, Laubacher, and Dellarocas, 2009]. All incentives should be applied with caution as rewards have been known to decrease annotation quality [Mrozinski, Whittaker, and Furui, 2008]. There are a number of common reasons to contribute to crowdsourcing projects which have been classified in different ways in the literature [Organisciak, 2015].

A classic distinction from the field of psychology is between *intrinsic* rewards (those that are internal to the user such as personal or social reward) and *extrinsic* rewards (those that are external to the user such as financial rewards) [Ryan and Deci, 2000], both of which are categorised as **internalisation** here. This distinction manifests itself in typologies of crowdsourcing systems as a distinction between paid and volunteer users [Geiger, Rosemann, and Fielt, 2011; Rouse, 2010; Schenk and Guittard, 2011]; however, these motivations may not be mutually exclusive [Mason and Watts, 2009]. The payment structure of extrinsic rewards, as well as the amount, may also have a impact on the ability to motivate a user [Aker et al., 2012; Geiger, Rosemann, and Fielt, 2011; Mason and Watts, 2009; Rokicki, Zerr, and Siersdorfer, 2015].

It is also important to distinguish between the motivation to participate (why people start doing something, a *primary motivation*) and the motivation to contribute (why they continue doing something, a *secondary motivation*) [Fenouillet, Kaplan, and Yennek, 2009; Organisciak, 2015], categorised as **continuation** here. Once both conditions are satisfied we can assume that a user will continue contributing until other factors such as fatigue or distraction break the cycle. This has been called **volunteer attrition**, in which a user's contribution diminishes over time [Lieberman, Smith, and Teeters, 2007].

Table 2.1: A table of common motivational reasons for participating and contributing in crowdsourcing, along with their classification, compiled from previous papers.

Reason	Internalisation	Motivation	Continuation
Interest in the topic	Intrinsic	Personal	Primary
Existing knowledge/opinions	Intrinsic	Personal	Primary
Ease of entry	Intrinsic	Personal	Primary
Ease of participation	Intrinsic	Personal	Primary
Novelty	Intrinsic	Personal	Secondary
Feedback and progression	Intrinsic	Personal	Secondary
Altruism and community	Intrinsic	Social	Primary
Sincerity and connection	Intrinsic	Social	Primary
Learning and reputation	Intrinsic	Social	Secondary
Social standing	Intrinsic	Social	Secondary
Support community	Intrinsic	Social	Secondary
Fixed fee	Extrinsic	Financial	Primary
Success-based (prize)	Extrinsic	Financial	Primary

By combining classifications of previous work a more complete picture of common motivations in crowdsourcing can be seen (Table 2.1).

Personal Incentives Personal incentives are evident when simply participating is enough of a reward for the user. Generally, the most important personal incentive is that the user feels they are contributing to a worthwhile project [Chandler and Kapelner, 2013]; however, personal achievement and learning can also be motivating factors.

People who contribute information to Wikipedia are motivated by personal reasons such as the desire to make a particular page accurate, or the pride in one's knowledge in a certain subject matter [Yang and Lai, 2010].

The opportunity to discover something unknown is a driving user motivation behind citizen science, such as image classification projects that use crowds to tag unknown objects leading to significant scientific discoveries [Clery, 2011]. The enthusiasm of the public to participate was most recently seen with the search for missing Malaysia Airlines flight MH370 in 2014 in which millions of users analysed satellite imagery,

tagging anything that looked like wreckage, life rafts and oil slicks.[1]

In contrast to previous classifications [Kaufmann, Schulze, and Veit, 2011] the idea of learning and education as a delayed payoff is considered an intrinsic, personal incentive here. The enjoyment of learning can be considered an incentive in itself; however, learning may also improve the chances of success with extrinsic factors such as career advancement or solving more complex tasks through a deeper understanding of the required knowledge [Brabham, 2012a].

Social Incentives Social incentives reward users by improving their standing amongst their fellow users and friends. By tracking the user's effort they can compete in leaderboards and see how their efforts compare to their peers. Assigning named levels for points awarded for task completion can be an effective motivator, with users often using these as targets, i.e. they keep working to reach a level before stopping, named the Zeigarnik effect [Rigby and Ryan, 2011].

News feed posts are a simple way users can make social interactions from an interface that is integrated into social networks such as Facebook or Twitter. Posting and sharing is an important factor in recruitment as surveys have shown that the majority of users participate because of a friend recommendation.[2] [3]

Financial Incentives Financial incentives reward effort with money. Direct financial incentives reward the user for the completion of a task or for successfully competing against other users (for example, achieving a high score). The former is the main method of motivating users of microworking systems, but a per-task reward may encourage users to manipulate the system, to do minimum work for maximum reward.

Indirect financial incentives reward the user irrespective of the work they have done such as entering each completed task into a lottery in which the winner is randomly selected (although doing more tasks would increase your chance of winning).

Whilst financial incentives seem to go against the fundamental idea behind GWAPs (i.e. that enjoyment is the motivation), it actually makes the enjoyment of potentially winning a prize part of the motivation. Prizes for high-scoring players will motivate hard-working or high-quality players, but the prize soon becomes unattainable for the

[1] http://www.tomnod.com/nod/challenge/mh370_indian_ocean

[2] http://www.infosolutionsgroup.com/2010_PopCap_Social_Gaming_Research_Results.pdf

[3] http://www.lightspeedresearch.com/press-releases/it's-game-on-for-facebook-users

majority of other players. By using a lottery-style financial prize the hard-working players are more likely to win, but the players who only do a little work are still motivated [Rokicki, Zerr, and Siersdorfer, 2015]. Whilst financial incentives are important to recruit new users, a combination of all three types of incentives is essential for the long term success of a project [Smadja, 2009].

Participation and workload Reported numbers of recruitment and participation mask a large disparity between how much contribution individual participants make. It is common for individual contributions to follow a Zipf power law distribution [Zipf, 1949], in which only a few users make the majority of the contributions. All users should be encouraged to contribute as the 'long tail' of collaborative data collection may account for as much as 30% [Kanefsky, Barlow, and Gulick, 2001].

A well-studied effect is called the Pareto Principle, in which 80% of the effects come from 20% of the causes [Pareto, 1896], or, in the context of crowdsourcing, 80% of the work is done by 20% of the people.

A similar proposal[1] is suggested in the 90-9-1 rule (or the 1% rule in Internet culture) that proposes that 1% of users create content (termed superusers), 9% edit or actively engage with content (termed contributors) with the final 90% of users doing nothing (termed lurkers) and has been shown to hold across a number of domains including social networks [van Mierlo, 2014]. In the context of crowdsourcing it could be suggested that 10% of the users contribute the majority of the work.

2.3.2 Evaluating users and annotations

Obtaining reliable results from non-experts is a challenge for crowdsourcing approaches, and in this context strategies for dealing with the issue have been discussed extensively [Alonso and Mizzaro, 2009; Alonso, Rose, and Stewart, 2008; Feng, Besana, and Zajac, 2009; Kazai, Milic-Frayling, and Costello, 2009].

The strategies for evaluating users and their annotations address five main issues:

1. Training users

2. Reducing genuine mistakes

[1] `http://www.nngroup.com/articles/participation-inequality`

3. Allowing for genuine ambiguity

4. Identifying outliers and cheating

5. Physical performance indicators

Training users A training stage is usually required for users to practise the task and to show that they have sufficiently understood the instructions to do a real task. The task design needs to correlate good user performance with producing good-quality data. The level of task difficulty will drive the amount of training that a user will need and the training phase has been shown to be an important factor in determining quality and improvement in manual annotation [Dandapat et al., 2009].

Simple tasks such as image tagging need very little instruction, whereas more complex judgements may require the users to be either more experienced or to undergo more training.

New users may initially perform badly but should improve with training and experience although lapses in concentration may still cause dips in performance. Training should engage the participant to increase their knowledge to become a pseudo-expert, i.e. the more they participate, the more expert they become. This graduated training makes a rating system (in which the user is regularly judged against a gold standard) essential to give appropriately challenging tasks. However, the distinction between experts and non-experts in the crowd may not be clear-cut [Brabham, 2012b].

Reducing genuine mistakes Users may occasionally make a mistake and press the wrong button. Attention slips need to be identified and corrected. The way the system is designed will effect how genuine mistakes can be corrected. In a collaborative system in which the users work openly together, they can correct their own, as well as others' mistakes. In a collective system, in which the users are working independently, a post-processing step is required to filter out mistakes from an otherwise competent user.

Allowing for genuine ambiguity Ambiguity is an inherent problem in all areas of language annotation [Jurafsky and Martin, 2008]. Resources should not only aim

to select the best, or most common, annotation but also to preserve inherent ambiguity, leaving it to subsequent processes to determine which interpretations are to be considered spurious and which instead reflect genuine ambiguity.

Identifying outliers and cheating Controlling cheating may be one of the most important factors in crowd-based system design. Several methods have been proposed to identify users who are cheating or who are providing spam annotations. These include checking the user's IP address (to make sure that one user is not using multiple accounts), checking annotations against known answers (the user rating system), preventing users from resubmitting decisions [Chklovski and Gil, 2005] and keeping a blacklist of users [von Ahn, 2006].

An additional method to evaluate the quality of the users is to use a multi-tier system in which one set of users reviews or rates the work of previous users [Quinn and Bederson, 2011], which is the fundamental idea behind the validation process (see Section 3.2.3).

A different approach is to identify those users who provide high-quality input. A knowledge source could be created based on input from these users and ignore everything else. Related work in this area applies ideas from citation analysis to identify users of high expertise and reputation in social networks by, for example, adopting the HITS algorithm [Yeun et al., 2009] or Google's PageRank [Luo and Shinaver, 2009].

Physical performance indicators The analysis of timed decision-making has been a key experimental model in cognitive psychology. Studies in Reaction (or Response) Time (RT) show that the human interaction with a system can be divided into discrete stages: incoming stimulus; mental response; and behavioural response [Sternberg, 1969]. Although traditional psychological theories follow this model of progression from perception to action, recent studies are moving more towards models of increasing complexity [Heekeren, Marrett, and Ungerleider, 2008].

It is possible to distinguish between three stages of processing required from the user to elicit an output response from input stimuli (see also Figure 2.2):

1. input processing (sensory processing) in which the user views the input (text or image) and comprehends it;

Figure 2.2: Stages of processing in human cognition.

2. decision making (cognitive processing) in which the user makes a choice about how to complete the task;

3. taking action (motor response) to enter the response into the system interface (typically using a keyboard or mouse).

This simple model demonstrates how a user responds to a task and can be seen in many examples of user interaction in task-based data collection systems. In crowdsourcing systems a user is given an input (typically a section of text or an image) and asked to complete a task using that input, such as to identify a linguistic feature in the text or to categorise objects in an image. The model can also be seen in security applications such as *reCAPTCHA*, in which the response of the user proves they are human and not an automated machine [von Ahn et al., 2008] and in users' responses to a search results page, with the list of results being the input and the click to the target document being the response [Macdonald, Tonellotto, and Ounis, 2012].

The relationship between accuracy in completing a task and the time taken is known as the Speed Accuracy Trade-off. Evidence from studies in ecological decision-making show clear indications that difficult tasks can be guessed when the costs of error are low. This results in lower accuracy but faster completion time [Chittka, Skorupski, and Raine, 2009; Kay, Beshel, and Martin, 2006]. Whilst studies using RT as a measure of performance are common, it has yet to be incorporated into more sophisticated models of predicting data quality from user behaviour.

2.3.3 Aggregating data

Once annotations have been collected from the crowd they need to be aggregated in some way to produce a best answer, or a set of plausible answers, to the task. The goal of aggregation is to use the contributions to approximate a single expert's answer, although crowd-created data allow for more complex probabilistic answer sets to be

created. Not all crowdsourcing systems need aggregation of the data due to the way they are produced, for example:

- **Individual**

  The one-user-one-idea individual method captures all input from users and treats each one as a separate solution. Typically each solution is then voted on by the crowd. Whilst this is not exactly an aggregation system it has been used for crowdsourcing ideas such as city planning.[1]

- **Consensus agreement**

  Given a set of solutions the users are required to come to a consensus regarding the best answer, typified by a court jury system. Wikipedia is also a form of consensus agreement in that the pages that are produced are edited until all the users agree it is appropriate coverage of a topic.

- **Peer prediction**

  The peer-prediction method is a recommender mechanism to motivate users to provide honest reviews of products [Miller, Resnick, and Zeckhauser, 2005]. The scheme uses one user's contribution to update the probability distribution of another user's answer and they do not score based on agreement but on the difference between the possible rating and the actual rating. The advantage of peer prediction is that it does not need any initiating gold standard and therefore is appropriate for assessing subjective opinions.

- **Find–Fix–Verify**

  The Find-Fix-Verify approach was first implemented in the crowd-based word processor called *Soylent* that enabled editing and summarising of text by the crowd [Bernstein et al., 2010]. The process breaks complex editing tasks into generative and review stages incorporating voting to produce a final result. In the *find* stage the users identify a section of text that needs work, in the *fix* stage users are asked to improve on the text and in the final *verify* stage the users vote on which improved text they prefer (or keep the original text).

[1] `http://ideascale.com`

A recent survey of crowdsourcing aggregation techniques defined them as either non-iterative or iterative processes [Hung et al., 2013]; the examples discussed here follow the same typology. Non-iterative aggregation uses methods to produce a score for each solution independently of the other tasks in the system. Examples of non-iterative aggregation include:

- **Averaging or median estimation** This method of extracting an answer from the crowd is based on the observation from Francis Galton in 1907 that the average (median) answer from the crowd could estimate the weight of a cow better than a cattle expert. This led to many different variations of crowdsourcing and answer aggregation using simple statistical methods [Surowiecki, 2005].

- **Majority voting** This is the idea that, given a finite set of things to choose from, the highest-voted is the best answer [Kuncheva et al., 2003]. The one person, one vote system is the foundation of many crowd systems as it is the most transparent and simple to implement. Repeated-labelling is a technique based on majority voting that takes uncertainty into account and is useful for estimating when an answer is good enough [Sheng, Provost, and Ipeirotis, 2008].

- **Condorcet voting** A less commonly used form of voting is Condorcet voting, in which the solutions to a task go through rounds of selection in order to reduce the number of possibilities until a best answer is found. Tournament selection and elimination selection are variations of this type of voting and have been shown to outperform majority voting on crowdsourced data [Sun and Dance, 2012].

- **Weighted voting** Weighted voting is similar to majority voting, but each vote is adjusted (or weighted) so that people who are most influential, most capable to answer or most popular (implemented differently in different systems depending on the output priorities) have more impact on the final decision. For example, the social honeypot method filters untrustworthy users in a pre-processing step by using trapping questions (for which the answer is already known) [Lee, Caverlee, and Webb, 2010]; this is a similar implementation to having a rating threshold for users based on their ability to perform tasks against a known gold standard. Another method of weighted voting is Expert Label Injected Crowd Estimation (ELICE) that uses the ratings of the users to judge the difficulty of the tasks,

thereby creating an object probability for the task [Khattak and Salleb-aouissi, 2011].

The superuser reputation scoring model in the social gaming network *Foursquare*[1] hints at the considerable commercial interest in weighting user contributions, and similar models are employed by other crowd-based datasets such as Stack Overflow [Bosu et al., 2013].

- **Directive models** The CrowdSense algorithm is a more complex version of weighted voting in which subsets of users are sampled based on an exploration/-exploitation criterion; the algorithm determines in real-time whether the system has collected enough data to produce a credible decision [Ertekin, Rudin, and Hirsh, 2014]. Similarly, a probabilistic model was developed from the *GalaxyZoo* data that use a set of Bayesian predictive models to make inferences as to how many users to direct to a task and what their abilities need to be in order to maximise efficiency of data collection [Kamar, Hacker, and Horvitz, 2012].

Iterative aggregation is more complex and performs a series of iterations over the data to adjust the value of each answer based on the expertise of the person who gave the answer, as well as measuring the expertise of the users from the available data; the process continues until there is convergence, i.e. no more changes are observed.

- **Expectation Maximization (EM)** The Expectation Maximisation algorithm iterates over the data, first by estimating the correct answer for each task using weighted voting (i.e. the skill of the user is taken into account) and then by estimating the quality of the users by comparing their answers with the inferred correct answer. This process continues until convergence in the data [Dawid and Skene, 1979]. Experiments with crowdsourced data have shown that it can be implemented in a straight-forward fashion and is flexible enough for most approaches [Ipeirotis, Provost, and Wang, 2010].

- **Probabilistic supervised learning** The probabilistic approach to aggregation is a similar method to Expectation Maximisation but characterises user ability by sensitivity (the ratio of correct positive answers) and specificity (the ratio

[1] `http://engineering.foursquare.com/2014/01/03/the-mathematics-of-gamification`

of correct negative answers). It shows significant ability to outperform simpler methods, although it is limited to binary data [Passonneau and Carpenter, 2013; Raykar et al., 2009, 2010]. A similar approach was used for estimating diagnostic accuracy in digital radiography [Albert and Dodd, 2008].

- **Generative Model of Labels, Abilities, and Difficulties (GLAD)** Another probabilistic approach, which is an extension of the EM approach, uses inference methods to infer simultaneously the ability of the user, the difficulty of the task and the probable label [Whitehill et al., 2009]. This model outperforms a majority vote method in both simulated and real crowdsourced data.

- **Iterative Learning** The Iterative Learning model operates in a similar way to EM, by estimating task difficulty and user ability; however, it treats each task and each user solution as separate instances of both, therefore a considerably more detailed view can be created [Karger, Oh, and Shah, 2011]. For example, the user's ability and bias over time can be observed and compensated for.

- **Multidimensional models** A multidimensional approach to classifying images models task characteristics in an abstract Euclidean space and each user is modelled as a multidimensional entity with variables such as competence, expertise and bias. The model can therefore cluster users and tasks based on these variables, not only to find tasks that are associated with each other, but also to discover schools of thought within the users [Welinder et al., 2010b].

2.4 Approaches to annotating data with a crowd

Several attempts have been made recently to bring order to the rapidly-developing field of collaborative creation on the Internet [Das and Vukovic, 2011; Malone, Laubacher, and Dellarocas, 2009; Quinn and Bederson, 2011; Wang, Hoang, and Kan, 2010; Yuen, Chen, and King, 2009]. The features of crowd-based approaches are discussing in more detail in Chapter 3; however, an introduction to each approach and notable efforts are presented here.

2.4.1 Peer production

Peer production is a way of completing tasks that relies on self-organising communities of individuals in which effort is coordinated towards a shared outcome [Benkler and Nissenbaum, 2006]. The willingness of Web users to collaborate in peer production can be seen in the creation of resources such as Wikipedia. English Wikipedia numbers (as of July 2015) 4,920,059 articles, contributed to by over 25.7 million collaborators.[1]

Wikipedia is perhaps the best-known example of peer production, but it is not an isolated case. *Open Mind Common Sense*[2], an artificial intelligence project whose goal was to construct a large commonsense knowledge[3] resource, demonstrated that Web collaboration can be relied on to create resources [Singh, 2002]. 14,500 volunteers have contributed nearly 700,000 sentences to *Open Mind Common Sense*, which has been turned into *ConceptNet*.[4] This is now one of the main sources of conceptual knowledge currently available.

A slightly different approach to the creation of commonsense knowledge with peer production has been pursued in the *Semantic MediaWiki* project [Krötzsch et al., 2007], an effort to develop a 'Wikipedia way to the Semantic Web', which aims to make Wikipedia more useful and to support improved search of Web pages using semantic annotation.

Peer production sites such as Wikipedia are now routinely used as a word sense repository [Csomai and Mihalcea, 2008] or as a source of encyclopedic knowledge [Ponzetto and Strube, 2007].

The key aspects that make peer production so successful are the openness of the data resource being created and the transparency of the community that is creating it [Dabbish et al., 2014; Lakhani et al., 2007].

Citizen science People who contribute information to Wikipedia are motivated by personal reasons such as the desire to make a particular page accurate, or the pride in one's knowledge in a certain subject matter [Yang and Lai, 2010]. This motivation is also behind the success of **citizen science** projects, such as the *Zooniverse* collection

[1] `http://meta.wikimedia.org/wiki/List_of_Wikipedias`

[2] `http://openmind.media.mit.edu`

[3] Commonsense knowledge are facts that an ordinary person is expected to know, such as a table has legs, a house has a roof, etc.

[4] `http://conceptnet.media.mit.edu`

of projects[1], in which the scientific research is conducted mainly by amateur scientists and members of the public [Clery, 2011]. The costs of ambitious data annotation tasks are also kept to a minimum, with expert annotators only required to validate a small portion of the data (which is also likely to be the data of most interest them).

Some citizen science projects get members of the public to classify objects in images taken from ROVs (Remotely Operated Vehicles)[2] [3] [4], whilst others require the users to supply the source data as well as the classification.[5] [6] [7] [8] The quality of citizen scientist generated data has been shown to be comparable to that generated by experts when producing taxonomic lists [Holt et al., 2013] even when the task is not trivial [He, van Ossenbruggen, and de Vries, 2013].

Citizen science efforts at annotating marine images show high accuracy for complex tasks, such as labelling and measuring scallops in images from *Seafloorexplorer*, with annotations correlating to expert annotations.[9]

Community Question Answering (cQA) Question answering systems attempt to learn how to answer a question automatically from a human, either from structured data or from processing natural language of existing conversations and dialogue. Here we are more interested in Community Question Answering (cQA), in which the crowd is the system that attempts to answer the question through natural language. Examples of cQA are sites such as StackOveflow[10], Yahoo Answers[11], Quora[12] and Github[13].

Image classification in a QA format is common in marine biology and SCUBA diving forums[14], but suffers from not having a broad enough community of users to answer the questions. Tasks on social networks follow a similar QA dialogue style in which

[1] https://www.zooniverse.org
[2] http://www.planktonportal.org
[3] http://www.seafloorexplorer.org
[4] http://www.subseaobservers.com
[5] http://www.projectnoah.org
[6] http://www.arkive.org
[7] http://www.brc.ac.uk/irecord
[8] http://observation.org
[9] http://blog.seafloorexplorer.org/2014/10/03/youre-doing-great-keep-it-up
[10] http://stackoverflow.com
[11] https://uk.answers.yahoo.com
[12] http://quora.com
[13] https://github.com
[14] http://www.scubaboard.com/forums/name-critter

threads may contain true tasks (when a question is asked and is answered) or implied tasks (when the post is augmented with additional data).

Detailed schemas [Bunt et al., 2012] and rich feature sets [Agichtein et al., 2008] have been used to describe cQA dialogue and progress has been made to analyse this source of data automatically [Su et al., 2007].

2.4.2 Microworking

Amazon Mechanical Turk[1] pioneered microwork crowdsourcing by using the Web as a way of reaching large numbers of workers (often referred to as turkers) who get paid to complete small items of work called human intelligence tasks (HITs). This is typically very little, in the order of 0.01 to 0.20 US$ per HIT.

Some studies have shown that the quality of resources created this way are comparable to that of resources created by experts, provided that multiple judgements are collected in sufficient number and that enough post-processing is done [Callison-Burch, 2009; Snow et al., 2008]. Other studies have shown that the quality does not equal that provided by experts [Bhardwaj et al., 2010] and for some tasks does not even surpass that of unsupervised language processing [Wais et al., 2010].

A reported advantage of microworking is that the work is completed very fast. It is not uncommon for a HIT to be completed in minutes, but this is usually for simple tasks. In the case of more complex tasks, or tasks in which the worker needs to be more skilled, e.g. translating a sentence in an uncommon language, it can take much longer [Novotney and Callison-Burch, 2010].

Whilst microworking remains a very popular crowdsourcing approach some serious issues regarding the rights of workers, minimum wage and representation have been raised [Fort, Adda, and Cohen, 2011]. Other microworking platforms, such as Samasource[2], guarantee workers a minimum payment level and basic rights.

Microwork crowdsourcing is becoming a standard way of creating small-scale resources, but is prohibitively expensive to create large-scale resources.

[1] `http://www.mturk.com`
[2] `http://samasource.org`

2.4.3 Gaming and games-with-a-purpose

Generally speaking, a game-based crowdsourcing approach uses entertainment rather than financial payment to motivate participation. The approach is motivated by the observation that every year people spend billions of hours playing games on the Web [von Ahn, 2006]. If even a fraction of this effort could be redirected towards useful activity that has a purpose, as a side effect of having people play entertaining games, there would be an enormous human resource at our disposal.

A game-with-a-purpose (GWAP) can come in many forms; they tend to be graphically rich, with simple interfaces, and give the player an experience of progression through the game by scoring points, being assigned levels and recognising their effort. Systems are required to control the behaviour of players: to encourage them to concentrate on the tasks and to discourage them from malicious behaviour.

GWAPs usually begin with a training stage for players to practice the task and also to show that they have sufficiently understood the instructions to do a real task. However, the game design must translate the task into a game task well enough for it still to be enjoyable, challenging and achievable. GWAPs need to correlate good performance in the game with producing good quality data.

Three styles of game scenario have been proposed for GWAPs [von Ahn and Dabbish, 2008]:

1. Output-agreement, in which the players must guess the same output from one input;

2. Inversion-problem, in which one player describes the input to a second player who must guess what it is;

3. Input-agreement, in which two players must guess whether they have the same input as each other based on limited communication.

The Output-agreement game scenario is the most straight-forward to implement and collect data from; however, other scenarios can make the game more interesting for the players and increase their enjoyment.

Structure of games GWAPs focuses on one main type of incentive: enjoyment. There are many reasons why people enjoy games (e.g. Koster [2005]) and models of enjoyment in games (called *the game flow*) identify eight criteria for evaluating enjoyment [Sweetser and Wyeth, 2005] (the model being based on a more generic theory [Csikszentmihalyi, 1990]):

1. Concentration - Games should require concentration and the player should be able to concentrate on the game;
2. Challenge - Games should be sufficiently challenging and match the player's skill level;
3. Player skills - Games must support player skill development and mastery;
4. Control - Players should feel a sense of control over their actions in the game;
5. Clear goals - Games should provide the player with clear goals at appropriate times;
6. Feedback - Players must receive appropriate feedback at appropriate times;
7. Immersion - Players should experience deep but effortless involvement in the game;
8. Social interaction - Games should support and create opportunities for social interaction.

The main method used by GWAPs to facilitate player enjoyment of the task is by providing them with a challenge. This is achieved through mechanisms such as requiring a timed response, keeping scores that ensure competition with other players, and having players of roughly similar skill levels play against each other.

Typically in a GWAPs a player can choose the type of task they find interesting and have some control over the game experience. Whilst some tasks are straightforward, others can provide a serious challenge. Players may also comment on the gaming conditions (perhaps to identify an error in the game, to skip a task or to generate a new set of tasks) and contact the game administrators with questions.

One of the simplest mechanisms of feedback is scoring. By getting a score the player gains a sense of achievement and some indication as to how well they are doing in the game.

GWAPs tend to be short, arcade-style games so immersion is achieved by progression through the game: by learning new types of tasks; becoming more proficient at current tasks; and being assigned a named level, starting from novice and going up to expert.

Serious games for learning GWAPs have a different goal to *serious games*, in which the purpose is to educate or train the player in a specific area such as learning a new language or secondary school level topics [Michael and Chen, 2005]. Serious games can be highly immersive, often in a 3D world, and have a directed learning path for the user as all of the data are known to the system beforehand. Therefore, the user can receive immediate feedback as to their level of performance and understanding at any point during the game.

GWAPs aim to entertain players whilst they complete tasks for which the system does not know, for the most part, the correct answer, and in many cases there may not even be a correct answer. Hence, providing feedback to users on their work presents a major challenge.

Gamification The concept of using game elements within a non-game context has a long tradition, but only recently has the term 'gamification' been defined [Deterding et al., 2011]. Feedback can be given to the user by tracking their performance in the system in order to encourage higher quantity or quality of work and motivational rewards can then be applied. Leaderboards and other comparative techniques show how well users are performing against their peers. User assessment in leaderboards can also be used as competency models, taking a multi-dimensional view of the user's abilities at different tasks [Seaborn, Pennefather, and Fels, 2013]. By using such methods, gamification aims to change the user's behaviour to meet the goals of the system designers [Zichermann and Cunningham, 2011].

Taken to its extreme, gamification becomes an approach more like GWAPs, in which the task is entirely presented as a gaming scenario rather than as a task with gaming elements applied.

GWAPs for image classification The GWAP approach showed enormous initial potential, with the first, and perhaps most successful, game called the *ESP Game*[1] attracting over 200,000 players who produced over 50 million labels [von Ahn, 2006]. In the game two randomly chosen players are shown the same image. Their goal is to guess how their partner will describe the image (hence the reference to extrasensory perception or ESP) and type that description under time constraints. If any of the strings typed by one player matches the strings typed by the other player, they both score points.

The quality of the labels has been shown to be as good as that produced through conventional image annotation methods. A GWAP approach to classifying images of wildlife (moths) called *Happy Moths* also showed good accuracy [Prestopnik, Crowston, and Wang, 2014].

Other image labelling games include the *Puzzle Racing* game [Jurgens and Navigli, 2014] and the two stage game (called *Infection* and *Knowledge Tower*) for validating image concepts [Vannella et al., 2014].

GWAPs for natural language processing *1001 Paraphrases* [Chklovski, 2005], one of the first GWAP the aim of which was to collect corpora, was developed to collect training data for a machine translation system that needs to recognise paraphrase variants.

The *Open Mind Common Sense* project also led to the development of a 'quasi-game' for collecting commonsense knowledge, the system *LEARNER* [Chklovski and Gil, 2005]. Other efforts to acquire large-scale world knowledge from Web users include *Freebase*[2] and *Evi* (formerly *True Knowledge*)[3].

Many of the ideas developed in *1001 Paraphrases* and *LEARNER*, are extremely useful, in particular the idea of validation.

Other GWAPs which have been used to collect data used in computational linguistics include:

- The GIVE games developed in support of the GIVE-2 challenge for generating instructions in virtual environments, initiated in the Natural Language Generation

[1] `http://www.gwap.com/gwap`
[2] `http://www.freebase.com`
[3] `http://www.evi.com`

community [Koller et al., 2010];

- The *OntoGame*, based around the *ESP Game* data collection model, aims to build ontological knowledge by asking players questions about sections of text, for example whether it refers to a class of object or an instance of an object [Siorpaes and Hepp, 2008];

- *JeuxDeMots* also aims to build a large lexico-semantic network composed of terms (nodes) and typed relations (links between nodes) [Lafourcade, 2007].

Several GWAPs have attempted anaphoric coreference such as *PlayCoref*, a two-player game in which players mark coreferential pairs between words in a text (no phrases are allowed) [Hladká, Mírovskỳ, and Schlesinger, 2009].

PhraTris [Attardi and the Galoap Team, 2010] is a GWAP for syntactic annotation using a general-purpose GWAP development platform called GALOAP.[1] *PhraTris*, based on the traditional game *Tetris*, has players arrange sentences in a logical way, instead of arranging falling bricks, and won the Insemtives Game Challenge 2010.

PackPlay [Green et al., 2010] was another attempt to build semantically-rich annotated corpora. The two game variants *Entity Discovery* and *Name That Entity* use slightly different approaches in multi-player games to elicit annotations from players. Results from a small group of players showed high precision and recall when compared to expert systems in the area of named entity recognition.

A more unified attempt at creating a gaming platform, named *Wordrobe*, targeted different linguistic tasks including part-of-speech tagging, named entity tagging, coreference resolution, word sense disambiguation and compound relations [Bos and Nissim, 2015; Venhuizen et al., 2013].[2] In addition to the suite of eight games players can choose between, it also offers a unique betting system allowing players to try to gain more points by indicating their confidence in their answer.

GWAPs have been used for other types of crowdsourced data collection [Thaler et al., 2011] including:[3]

- Video annotation such as *OntoTube*, *PopVideo*, Yahoo's *VideoTagGame* and *Waisda*;

[1] `http://galoap.codeplex.com`

[2] `http://www.wordrobe.org`

[3] See the Appendix A for a list of GWAPs and where they can be found.

- Audio annotation such as *Herd It*,*Tag a Tune* and *WhaleFM*;
- Biomedical applications such as *Foldit*, *Phylo* and *EteRNA*;
- Transcription such as *Ancient Lives* and *Old Weather*;
- Acquiring commonsense knowledge such as *Verbosity*, *OntoGame*, *Categorilla* and *Free Association*;
- Improving search results such as Microsoft's *Page Hunt*;
- Social bookmarking such as *Collabio*;
- Changing human behaviour such as *Power House*.

2.4.4 Social computing and social networks

Social computing has been described as 'applications and services that facilitate collective action and social interaction online with rich exchange of multimedia information and evolution of aggregate knowledge' [Parameswaran and Whinston, 2007]. It encompasses technologies that enable communities to gather online such as blogs, forums and social networks, although the purpose is largely not to solve problems directly.

Here we make a distinction between using a social network as a platform to deploy a system compared to using the social network itself as the platform for problem solving.

Deploying systems on social networks In recent years, social networking has become the dominant pastime online. As much as 22% of time online is spent on social networks such as Facebook, Twitter and others. This is three times the amount of time spent emailing and seven times the amount of time spent searching the Internet.[1]

The success of social network games such as *Candy Crush Saga*, with 150 million active players each month, show the potential for large-scale participation using social networking platforms.[2] An estimated 927 million hours are spent each month by Facebook users playing games[3], which is another indicator of the vast human resource available.

[1] `http://mashable.com/2010/08/02/stats-time-spent-online`
[2] `http://appstats.eu` (accessed Feb 2013)
[3] `http://www.allfacebook.com/facebook-games-statistics-2010-09`

A study of US and UK social network users showed that Facebook was by far the most frequently used platform for social network gaming (used by 83% of users, compared to MySpace, the next highest platform also used, at 24%).[1]

GWAPs integrated into social networking sites such as *Sentiment Quiz* [Rafelsberger and Scharl, 2009], *Rapport Game* [Kuo et al., 2009] and *TypeAttack* [Jovian and Amprimo, 2011] on Facebook show that social interaction within a game environment does motivate users to participate. Another Facebook GWAP that validated automatically extracted common sense knowledge was the *Concept Game* [Herdagdelen and Baroni, 2012].

DigiTalkoot's games *Mole Hunt* and *Mole Bridge*, released on Facebook by the National Library of Finland and Microtask to help digitise old Finnish documents, attracted 110,000 participants who completed over eight million word-fixing tasks in 22 months.[2]

It is unclear whether socially networked games change the dynamic of user types, such as the suggestion that players can be categorised in four types: killers; acheivers; explorers; and socialisers [Bartle, 1996].

Inherent problem solving on social networks The open dialogue and self-organising structure of social networks allow many types of human interaction, but here we are most interested in the idea of community problem solving, in which one user creates a task and the community solves it for them. A common task is to identify something in an image.

Facebook has a vast resource of uploaded images from its community of users, with over 250 billion images, and a further 350 million posted every day. Images of things (rather than people or places) that have been given captions by users only represent 1% of these data, but it is still of the order of 2.6 billion images.[3]

As social networks mature the software is utilised in different ways, with decentralised and unevenly-distributed organisation of content, similar to how Wikipedia users create pages of dictionary content.

Increasingly, social networks are being used to organise data, to pose problems, and to connect with people who may have solutions that can be contributed in a simple and

[1] http://www.infosolutionsgroup.com/2010_PopCap_Social_Gaming_Research_Results.pdf

[2] http://www.digitalkoot.fi/index_en.html

[3] http://www.insidefacebook.com/2013/11/28/infographic-what-types-of-images-are-posted-on-facebook

socially-convenient fashion. Facebook has been used as a way of connecting professional scientists and amateur enthusiasts with considerable success [Gonella, Rivadavia, and Fleischmann, 2015; Sidlauskas et al., 2011]. However, there are drawbacks with this method of knowledge sharing and problem solving: data may be lost to people interested in them in the future and they are often not accessible in a simple way, for example, with a search engine.

2.5 Summary

This section discussed related work to harnessing collective intelligence on social networks, firstly by detailing prior art in the problem space of text annotation and image classification, then discussing how previous work has attempted to use a crowd to solve the problem. Outsourcing tasks to distributed humans, termed crowdsourcing, creates some interesting problems of motivation, incentivisation, quality control and choice of the best answer. The next section explores common features of crowdsourcing, in particular whether social networks fit within this scheme.

3

Models for harnessing collective intelligence

Chapter 2 discussed several crowd-based approaches that can be used to replace the traditional expert-annotator model. This section abstracts key features from each approach and discusses using an additional stage to data collection, namely to have the workers perform both the task of providing the judgements (annotations) and the task of checking those judgements (validation).

The primary research question of this thesis is whether collective intelligence on social networks can be used to create large-scale **data** resources, with high-quality labelling of **information** about the data, that can be used to create **knowledge** to solve problems that cannot be addressed in any other way yet. This question makes one important assumption: that social networks can be viewed as problem-solving systems. If this assumption holds true then a wealth of ideas and research regarding crowdsourcing and collective intelligence analysis is at our disposal. This chapter investigates this hypothesis by comparing social network systems to other crowdsourcing approaches using a set of common features.

In this chapter a number of concepts are specifically defined that relate to the features of models but can have different meanings associated within each approach. A **task** is a construct that has a problem presented via a system and a methodology is followed to arrive at a solution (or set of solutions). The person, agent or agency

Portions of this chapter previously appeared in Chamberlain, Kruschwitz, and Poesio [2012]; Chamberlain [2014b,c].

that creates the system and uses the final aggregated output is called the **requester** and a person or agent that contributes to the solution of the task is called a **worker**. A contribution from a worker can be an annotation or validation; both are collectively described as **work**. Each unique solution to a task is called an **interpretation**. The collection of interpretations is called the system **output**.

3.1 Features of annotation models

Crowdsourcing approaches can be distinguished by a number of common features related to the data, task, worker and output of the system and present their own challenges. Several reviews of features have been presented in the literature in relation to different information science fields, either with the aim of classifying existing work or to identify new areas of crowdsourcing. Previous work typically focuses on the type of task, quality control, user motivation and aggregation [Brabham, 2013; Das and Vukovic, 2011; Geiger, Rosemann, and Fielt, 2011; Malone, Laubacher, and Dellarocas, 2009; Organisciak, 2015; Quinn and Bederson, 2011; Rouse, 2010; Schenk and Guittard, 2011]. The aim here is not to repeat the existing work nor to build a complete facet set for crowdsourcing; rather it is to focus on the features that are of most importance in the context of distributed human intelligence tasks (see Section 2.3). Each feature is discussed in relation to previous work and summarised in Tables 3.1, 3.2, 3.3 and 3.4.

Generalised crowd approaches to problem solving are discussed for the purposes of exploring the features that are common between systems. There will always be exceptions to the generalisations and the purpose here is not to pigeon-hole research, but to see where specific work overlaps on the continuum of these ideas. To clarify why these features apply to a particular approach an exemplar system is chosen for the approach that is perhaps the most prevalent or successful. For manual expert annotation, the traditional methodology outlined in Section 2 is used; for peer production, *GalaxyZoo* represents citizen science (although a detailed typology for citizen science projects also exists [Wiggins and Crowston, 2011]), StackOverflow represents Community Question Answering (cQA) and Wikipedia's main website is an example of a wiki-type approach (see Section 2.4.1); for microworking, Amazon's Mechanical Turk outlined in Section 2.4.2 is used; for GWAPs, the *ESP game* outlined in Section 2.4.3 is used and finally

Table 3.1: A table showing data features, including who creates the data and who manages the tasks.

Approach	Data creation	Task management
Expert annotation	Requester	Requester
Peer production: Citizen science	Requester	Requester
GWAP	Requester	Requester
Microworking	Requester	Requester
Peer production: Wikipedia	Worker	Worker
Peer production: cQA	Worker	Worker
Social Networks	Worker	Worker

for social networks, Facebook itself is considered (rather than a system implemented on the platform, see Section 2.4.4).

3.1.1 Data features

Data creation Most studies in crowdsourcing use the paradigm of a requester having a collection of data that they require to be annotated. However, in some projects it is the workers themselves that create small amounts of data on which they want a task completed and the requester accesses both the submitted data and the subsequent output. This is typical for a citizen science or social networking approach in which the worker who sets up the task also provides the data for the task to be solved (for example, posting an image that needs to be identified). This feature is rarely mentioned in the literature as the assumption is that the requester provides the data to work on; however, the idea of *generative* (workers create the data) vs *reactive* (workers react to data) tasks has been proposed as a feature [Schenk and Guittard, 2011]. Whilst the paradigm of getting workers to submit data can be very powerful it also adds a further motivational burden on the system, namely how to get the workers to submit data in addition to providing the work (see Table 3.1).

Task management The management of the system covers the dimensions of *who* uses crowdsourcing and *why* crowdsourcing systems exist [Malone, Laubacher, and Dellarocas, 2009]. Management is largely dependent on the desired output of the requester; however, it is not always the case that the data and tasks are fully managed and this

Table 3.2: A table showing task features, including whether the input is constrained, in what order it can be entered and who checks it.

	Input constraint	**Input order**	**Validation by**
Expert annotation	Constrained	Both	Requester
Peer production: Citizen science	Constrained	Parallel	Requester
GWAP	Constrained	Parallel	Both
Microworking	Constrained	Parallel	Requester
Peer production: Wikipedia	Unconstrained	Series	Worker
Peer production: cQA	Unconstrained	Series	Worker
Social Networks	Unconstrained	Series	Worker

can be left to the workers, although at the risk of an unbalanced output dataset. For example, there is no central control as to what Wikipedia pages should be created and how much content should be contributed. Popular subjects such as entertainment and celebrities have considerably more content than other subjects. However, there are ways for tasks to be implied, such as by creating a link to a page that does not currently exist (these are highlighted in red on Wikipedia and lead to a 'Create a page for this subject' template). Similarly, on social networks there may only be an implication of what is required of the community and the content and tasks that are added are decided upon by the workers (see Table 3.1).

Task management is a similar concept to the **director** feature in which tasks can be *sponsored* (have a requester pushing the task) or *autonomous* (when the tasks are self-generated) [Zwass, 2010]. However, this masks the distinction between a requester who drives a task (sponsored) and is also a worker in the crowd (autonomous).

When there is worker-managed task creation, it is intuitive to think that workers would add harder tasks because simple tasks would either have already been done (such as popular Wikipedia pages) or are not worth putting on the system (such as images that can be classified easily by the requester).

3.1.2 Task features

The type of task that is presented covers the dimension of *how* the problem gets solved [Malone, Laubacher, and Dellarocas, 2009]. One of the important features for distin-

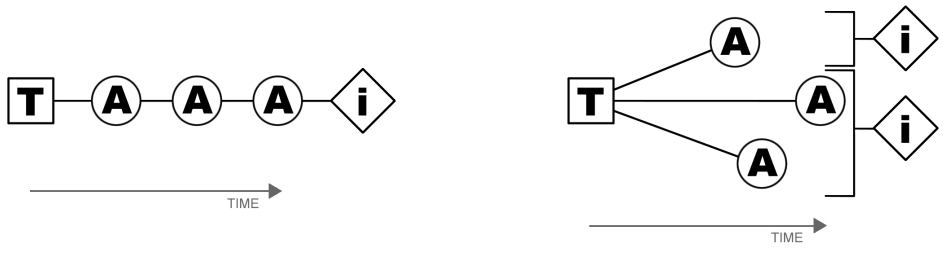


Figure 3.1: A task T can be completed in series (left) in which each annotation A is dependent on the one before and leads to one interpretation i (Wikipedia, cQA and social networks). Alternatively T can be completed in parallel (right) in which annotations can be entered simultaneously leading to multiple interpretations that require post-processing for a final output (microworking, GWAPs and traditional expert annotation).

guishing individual projects (rather than the approach) is to look at **task difficulty**, either as a function of the task (*routine, complex* or *creative* [Schenk and Guittard, 2011]) or as a function of worker *cognitive load* [Quinn and Bederson, 2011]. Also useful for distinguishing between projects is the **centrality** of the crowdsourcing in the system, i.e. is the crowdsourcing *core* to the system, such as creating content in Wikipedia, or is it *peripheral* such as rating articles [Organisciak, 2015].

Input constraint Whilst data are often structured, mainly to allow them to be input into the system, the annotations may not necessarily be. Crowdsourcing typically constrains workers to enter a restricted range of inputs via radio buttons and dropdown lists, whereas social networks and peer production allow unconstrained text input that requires post-processing. Some tasks require annotations to be aligned to an ontology and this provides structure; however, spelling mistakes and ambiguity can cause errors. Along with unconstrained page creation, Wikipedia allows for semi-constrained input through summary boxes on each page (see Table 3.2).

The choice of input constraint may be driven by a further facet of whether the answers to the task need to be *objective* or *subjective* [Organisciak, 2015].

Input order The timing of the presentation of the tasks is dependent on the system and, generally speaking, will determine how fast a system can produce an output for a task. In the case of Wikipedia, cQA and social networks, a task is added and each worker contributes in series, i.e. each contribution is dependent on the previous contributions in the way a Wikipedia page is developed or a conversation thread flows. Workers on Wikipedia can edit and overwrite the text on a page. This 'last edit wins' approach is fundamental to building the content; however, contentious subjects may cause 'edit wars' and pages may become locked to prevent future editing.

In order to increase the crowdsourcing efficiency, some systems allow tasks to be completed in parallel, i.e. multiple workers annotate different tasks at different times meaning that not all tasks will be completed in the same amount of time (see Figure 3.1). Parallel tasks are common in microworking, GWAPs and citizen science. Expert annotation can be completed both in series or in parallel (see Table 3.2).

A wider, systematic view of task order would be to view the system's **procedural order** and how the worker interacts with system inputs and responses from the crowd [Organisciak, 2015].

Validation Quality control of a system is a feature of most typologies of crowdsourcing and can be used to distinguish between different projects [Das and Vukovic, 2011; Quinn and Bederson, 2011]; however, it creates a large and complex facet group that is beyond the scope of what is required here. In this context, it is the reviewers of the annotations supplied by the workers that is of interest.

Validation on some level occurs after annotations have been applied to the data; the issue is whether those validations are part of the process that the workers are involved in or whether it is a form of checking from the requester to ensure that a sample of the annotations are of a high enough quality. It is typically the case for requesters to check a sample of annotations with experts, microworking and citizen science. In systems such as Wikipedia, social networks and cQA, the checking and validation of all answers is done by the workers themselves. GWAP annotations are typically validated by the requester; however, an increasing proportion of games are using validation as an additional worker task to reduce the workload for the requester (see Table 3.2).

Table 3.3: A table showing worker (user) features, including how they are motivated, trained and work together.

	Worker motivation	**System training**	**Group working**
Expert annotation	Money	Explicit	Collective
Peer production: Citizen science	Personal	Explicit	Collective
GWAP	Personal/Social	Explicit	Collective
Microworking	Money	Explicit	Collective
Peer production: Wikipedia	Personal	Social	Collaborative
Peer production: cQA	Personal	Social	Collaborative
Social Networks	Personal/Social	Social	Collaborative

3.1.3 Worker (user) features

Worker motivation One of the most serious failings of collective intelligence systems is the lack of participation and so a key feature is the motivation of the workers, corresponding to the dimension of *why* users would participate in crowdsourcing [Malone, Laubacher, and Dellarocas, 2009]. Incentives are commonly divided into personal, social and financial categories (for a more complete discussion of motivation, see Section 2.3.1). Expert and microworking workers are primarily motivated by financial rewards, although they may also gain personal satisfaction from being part of a project. Peer production workers are typically driven to participate because of an altruistic desire to help the project. Workers on GWAPs are driven by the enjoyment of playing the game, which is a complex combination of personal and social incentives (discussed in Section 2.4.3). Workers on social networks also have a complex combination of personal and social motivations.

Citizen science, GWAPs and microworking are all established methods of replicating an expert's effort at solving tasks and the main issue is how to motivate the worker to complete tasks to a high quality and quantity. This approach has been referred to as 'chocolate covered broccoli', an analogy for making workers do something they normally wouldn't do by rewarding them [Bruckman, 1999]. Peer production such as Wikipedia, cQA and social networks are complex personal and social reward systems in which the worker is participating because it is part of what they are trying to otherwise achieve (see Table 3.3).

System training In order for the workers to create annotations they must learn how to use the interface and to understand the task. Both of these training needs can be addressed explicitly by providing the workers with written instructions, walk-through demonstration tasks and a sample set of data in which directed feedback is provided. However, some systems, notably Wikipedia, cQA and social networks, only have minimal (if any) instructions on how to use the system or how to complete the task. Workers in these systems observe the behaviour of other workers: how they create and solve tasks and the degree of quality that is expected. This 'lurking' behaviour is often portrayed as a negative aspect of Internet culture (see Section 2.3.1) but, in terms of the worker gaining an understanding of the task, it is actually a vital part of social training. Social networks in particular also benefit from the worker already knowing how to participate in the system as they will have learnt to post messages and replies in other forms of interaction and there is only a small additional requirement to learn how to interact with the task, for example using a Twitter hashtag handle or posting relevant additional metadata required to solve the task (see Table 3.3).

How the worker is trained is an issue not covered in other typologies, most likely because the training in a system is viewed as a supplement, rather than a distinguishing feature. The closest feature mentioned in other work is the idea of **worker investment** in terms of the amount of time a worker must spend learning the system before they can use it [Quinn and Bederson, 2011].

Previous typologies have used **pre-existing skills** as a way to define projects, but it is very complex, if not impossible, to classify workers generally in this way. For example, there is a distinction between *unskilled, locally trained* and workers with *pre-existing knowledge* [Organisciak, 2015]; however, the systems seen in practice show that workers are on a continuum of learning and their ability to answer tasks of anything more than the most trivial type will improve over time, based on the level of task complexity that is allowed into the system. As an example, a worker contributing to Wikipedia could be using their pre-existing knowledge to add content to the page, or use the knowledge they have learnt on similar topics, or simply to edit the grammar or spelling. Where the crowd came from (or **crowd type**) has also been considered a feature, whether they are a *closed, internal community* or an *open, external community* (or both) [Das and Vukovic, 2011]. This is a useful distinction within information science but not useful in this context.

Related to the idea of pre-existing knowledge and source of the workers is the feature of worker **diversity**. This is a difficult feature to determine in most systems; however, it is a useful consideration for what type of answers are required, with *diverse* groups likely to provide multiple answers and *homogeneous* groups likely to work towards a consensus on the best answer [Organisciak, 2015].

Group working GWAP and microworking approaches typically have tasks that workers perform on their own and this is designed into the system to prevent collusion to gain rewards or prevent copying of the most common answer. Whilst collective systems such as these ensure each annotation is not biased at the time of submission they may restrict a human's ability to perform complex tasks. Allowing workers to collaboratively work together in groups in which they can see each other's annotations may become biased towards a particular answer (which may or may not be a good thing). This may be because a trusted worker has suggested the annotation or because workers that might disagree are reluctant to make alternative annotations when there is majority agreement. However, the social aspects of collaboration, such as feeling part of a group and making friendships (on a superficial level at least) are powerful motivators. Some citizen science systems combine both paradigms by getting workers to work individually on tasks, but allowing the answers to be posted to a forum for discussion if something interesting or challenging is found (see Table 3.3).

How the workers work together to complete the task is a part of a larger feature described as **aggregation** in the literature in order to distinguish projects. As is apparent from the related work, most systems will deploy a variety of techniques to aggregate answers, either as a strategy for workers to enter work or to post-process the work to remove poor quality and identify outliers. A useful approach to classifying aggregation can be seen with *integrative* (data are pooled to a common resource) vs. *selective* (data are combined to find a best answer) classification [Geiger, Rosemann, and Fielt, 2011; Schenk and Guittard, 2011], although a more complex classification has been proposed of *summative, iterative* and *averaged* aggregation [Organisciak, 2015].

3.1.4 Output (implementation) features

Beneficiary Crowdsourcing systems are typically seen as a way to get a task from the requester completed using a set strategy and the data that are created is only of

Table 3.4: A table showing output (implementation) features, including the beneficiary, the accessibility of the data and whether the worker receives recognition.

	Beneficiary	**Output accessible**	**Worker recognition**
Expert annotation	Requester	No	No
Peer production: Citizen science	Requester	No	Yes
GWAP	Requester	No	No
Microworking	Requester	No	No
Peer production: Wikipedia	Worker	Yes	No
Peer production: cQA	Worker	Yes	Yes
Social Networks	Worker	Yes	Yes

direct benefit to the requester [Rouse, 2010]. However, in some peer production and social network systems it is the worker who creates the task and also the worker who benefits from the task being completed. For example, a Wikipedia worker might create a page on a topic they are interested in, which creates an implied task for other workers to enter more information on the topic. The original worker then benefits from having a much larger page of information created by other workers. On social networks and cQA systems it is a worker who posts a task and directly benefits from the task being solved (see Table 3.4).

A different way to express this feature is to define it by **task request cardinality**, such as *one-to-one*, in which the one worker completes one task (such as expert annotation), *many-to-one*, in which the crowd provide an answer for the requester (such as microworking), or *many-to-many*, in which the crowd create a resource for the crowd (such as Wikipedia) [Quinn and Bederson, 2011].

Output accessible Another feature of systems is who can access the final output data. Typical crowdsourcing projects do not allow access to the output dataset, although a proportion of it may be shared in the long term for scientific research projects. However, with some peer production and social network systems the output dataset is open and accessible from the point of data entry. For example, Wikipedia workers can see the page edits immediately and this information can be accessed directly. Similarly, social network workers can see the data being entered directly and this can be searched

and accessed (see Table 3.4).

Worker recognition The final feature of systems is whether the worker gets recognition for their efforts, which has also been included in other taxonomies [Quinn and Bederson, 2011]. The worker must be identifiable on the system and across tasks in order to build a reputation within the community. Contribution to science, learning and discovery are the driving motivations behind citizen science participation [Raddick et al., 2008]. Worker recognition can be taken to extremes when new knowledge is found, such as the naming of newly discovered objects[1] (see Table 3.4).

3.2 The Annotation Validation (AV) Model

The evaluation of features of crowdsourcing approaches (Section 3.1) shows that there are overlapping ideas that can be applied in different ways. These generalisations have exceptions and many systems do not conform to this typology as developers and researchers look across to other approaches to improve and develop their systems.

One feature that has recently been applied across approaches is to use the workers to perform the checking of annotations in a so-called **validation mode**. In the validation task the worker sees the interpretations from the previous worker(s) and agrees with it or not.

3.2.1 Annotations: How many do you need?

Researchers investigating single-tier crowdsourcing systems, typified by microworking, make the assumption that if an answer is possible from the crowd then getting lots of annotations, whilst applying filtering, will eventually lead to the best answer [Snow et al., 2008]. In some cases this may prove to be the case; however, the caveat of getting more annotations is the chance of getting a more diverse range of answers or noise, from which the true answer cannot be extracted. It may be that there is no best solution to the task and no amount of additional annotations will lead to a best answer.

The basic statistical probability of getting a correct interpretation given a number of annotations shows that the worker rating (the assessed ability of the worker to provide the correct answer) will determine how many annotations you might need per task (see

[1] `http://www.universetoday.com/82358/hubble-eyes-hannys-voorwerp`

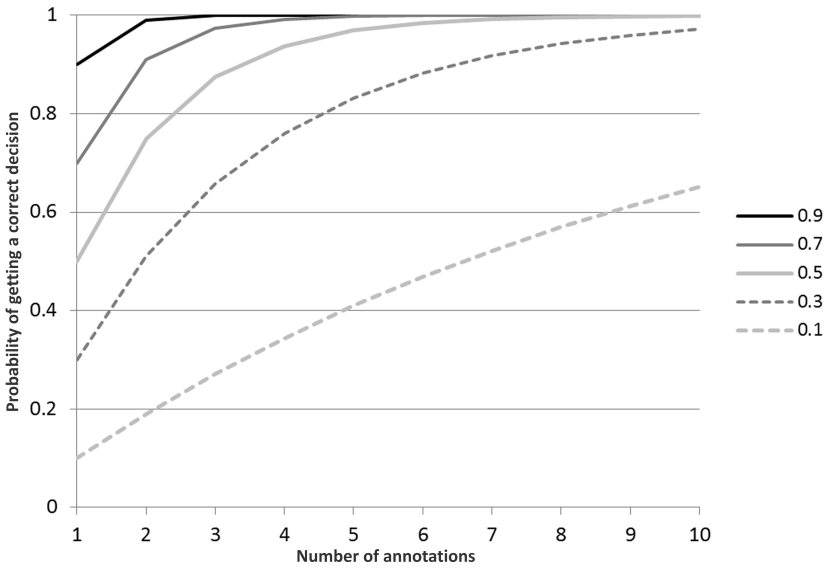


Figure 3.2: Chart showing the chance of getting a correct decision with each additional annotation for different levels of worker rating.

Figure 3.2). If we require a 99% probability of getting a correct interpretation from the workers and if each worker has a 90% chance of submitting a correct answer, only two annotations are needed. If the workers' ratings are less, say 70% chance, then four annotations are needed, and if less again at 50% then seven annotations are needed. A crowd with an average lower than 50% chance will take considerably more annotations.

This naive model does not account for the variability in player abilities, the order in which players of different abilities submit answers, the difficulty of the task, the possibility of having multiple correct answers or other confounding factors. It also offers no way of identifying the correct answer from the submissions. It is important to estimate the number of annotations that are required; too few annotations and the correct interpretation for the task might not be discovered; too many annotations and the data collection will take longer than necessary, cost more (if using financial rewards) and introduce more noise (incorrect interpretations) that need to be filtered out.

3.2.2 Supporting annotation with validation

The fundamental idea behind using validation as a supporting mechanism for annotation is that it should be easier and faster for the worker to decide if an interpretation is correct rather than create an interpretation as an annotation. In one sense an agreeing

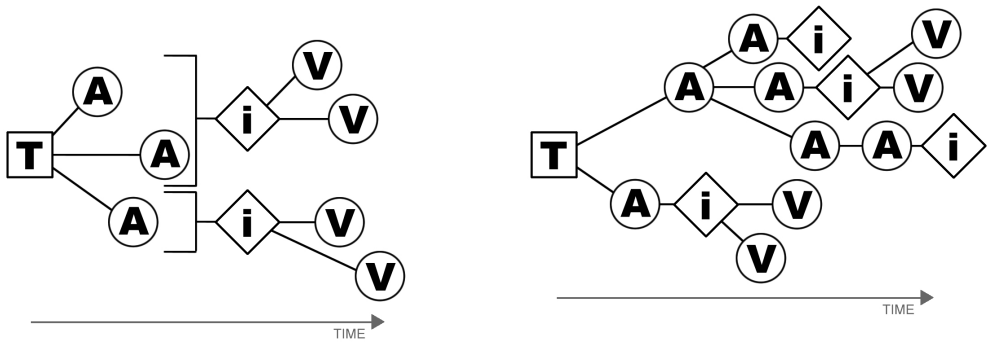


Figure 3.3: Validation (V) of interpretations (i) can be completed synchronously (left) in which the annotation (A) stage is completed before the task (T) is presented in validation mode, as with the AV Model implemented in the GWAP in Section 3.2.3, or asynchronously (right) in which the annotation and validation stages occur simultaneously, such as social networks or cQA systems.

validation can be seen as another annotation in favour of the interpretation (if using a majority voting count to determine the best answer). A disagreeing validation on the other hand provides less information, in that the worker is saying what the correct interpretation is not, rather than what it is. An agreeing validation says what the interpretation is and by inference what it is not (if we assume there is only one correct or best answer).

A validation step can be added in two ways: either synchronous, in which validation is completed after an initial annotation stage is complete, which is the case for the AV Model implemented in the GWAP discussed in Section 3.2.3, or asychronous in which the task is annotated and validated together, such as a conversation thread on cQA or social networks (see Figure 3.3).

3.2.3 Evaluating workers and their contributions

The AV Model can be implemented in a system to provide feedback on worker performance. In an annotation-only system workers can only be rewarded for quantity, not quality, which is typical for microworking. When the interpretations are not known beforehand a system using validation can provide feedback to the worker on their performance based on how much they agree with other workers.

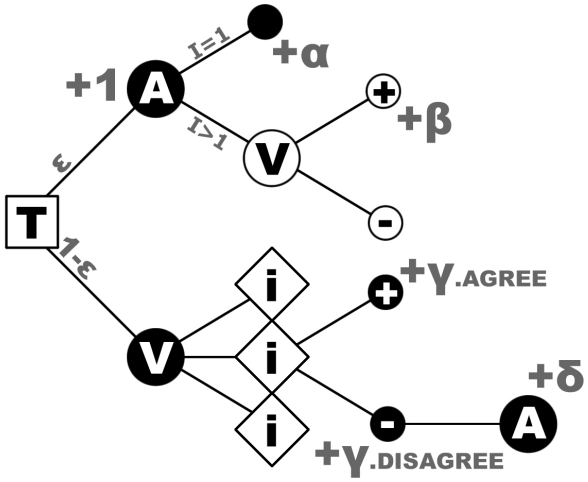


Figure 3.4: A representation of the AV Model showing how a worker's score is calculated for a task (T) in either Annotation Mode (A) or Validation Mode (V). Black circles indicate a worker input and white circles indicate an input created by other workers in the system.

The AV Model describes a shift from effort-based reward, in which the reward is proportional to the number of tasks completed irrespective of the quality, to agreement-based reward in which the workers receive more reward for higher quality (or more commonly agreed with) solutions.

There are three key benefits the validation process offers:

1. to reward workers appropriately for solutions to tasks without assessing quality with a gold standard;

2. to assess worker ability by predicting their response to the tasks;

3. to filter a noisy dataset with post-processing.

3.2.4 Description of the AV Model

This section describes the algorithm behind the AV Model, see Figure 3.4 for a diagrammatic representation.

Initially workers complete annotation tasks (Annotation Mode) and are given a fixed reward for their contribution. If the initial group of workers (U_A) enter the same solution they are all rewarded again (α); however, it is likely they will create multiple

interpretations (I) for the task. In the latter case each interpretation is presented to further workers (U_V) in a binary (agree or disagree) validation task (Validation Mode).

$$\alpha = P_u P_{ub}^{U_A-1} + \frac{(1-P_u)(1-P_{ub})^{U_A-1}}{(I-1)^{U_A-2}} \tag{3.1}$$

The validating worker is rewarded for every annotating worker that they agree with (γ). If they disagree with the interpretation they receive a reward for every annotating worker that entered a different interpretation to the one presented, hence they must also be disagreeing.

$$\gamma = \frac{U_A(P_u P_{ub} + 2(1-P_u)(1-P_{ub}) + P_u P_{ub}(I-1) + (1-P_{ub})(I-2)}{I} \tag{3.2}$$

If the validating worker disagreed with the interpretation they are asked to enter an interpretation using annotation and are rewarded again for their contribution (δ).

$$\delta = \frac{1 - P_u + P_u(I-1)}{I} \tag{3.3}$$

If the worker creates a new interpretation this will also be validated. Every time a validating worker agrees with an interpretation, any worker from the original annotating group that entered the interpretation will also receive a retrospective reward (β).

$$\beta = U_V(1-\alpha)(P_u P_{ub} + (1-P_u)(1-P_{ub})) \tag{3.4}$$

Additionally, P_u is the probability that the worker selects the correct answer (also called the rating) which is calculated by giving the worker a small set of tasks with a known answer; P_{ub} is the mean probability of a worker in the system (the user base) selecting the correct answer. ϵ is the proportion of tasks presented in an annnotation task, which is an estimation of data maturity, and S is the predicted score per task for the worker.

$$\epsilon = \frac{U_A}{U_A + U_V I} \tag{3.5}$$

$$S = \epsilon(1 + \alpha + \beta) + (1-\epsilon)(\gamma + \delta) \tag{3.6}$$

The model makes several assumptions:

- I is greater than 1;

- there is only one correct interpretation per task;
- the worker will try to solve the task by choosing the correct interpretation;
- the worker only sees the task once.

Whilst hypothetically possible to have a value of I=1, i.e. only one interpretation per task, there would be no value in using a system like this as all the workers would enter the same decision, either because the task is very easy or the workers are very good.

The model assumes there is only one correct interpretation, but in the case of linguistic analysis, relevance judgement and many other applications there is likely to be more than one possible interpretation and the model should be extended to accommodate multiple correct interpretations. Adding interpretations after the initial group of workers have submitted their annotations allows the system to capture less popular solutions and avoid convergence, in which workers choose what they think will be a popular solution, rather than the best solution.

It is assumed that the worker will always try to select the best solution, but this is clearly not the case for some workers who employ strategies to maximise rewards for minimum effort. There are numerous ways a worker can manipulate a system to their advantage and it is the job of system designers to minimise this impact, either at the moment of entering the data or in post-processing.

One cheating strategy is to enter the fastest and most predictable combination of inputs in order to gain points by quantity rather than quality. Post-processing of these noisy data are required by looking at performance measures such as the time to complete a task (see Section 2.3.2). There is also the possibility that workers can collude in their answers as it is in their best interest to agree with each other. This is one reason why one would use a collective system over a collaborative system (see Section 3.1.3).

The model assumes that the worker only receives the task once, in either mode, but this may not be the case. Workers may occasionally be given the same task (although not necessarily in the same mode) to measure implicit agreement, i.e. the probability the worker will provide consistent results. The worker's ability should improve over time so they may provide different, higher-quality interpretations to tasks they have done before and this could be used to normalise their result set.

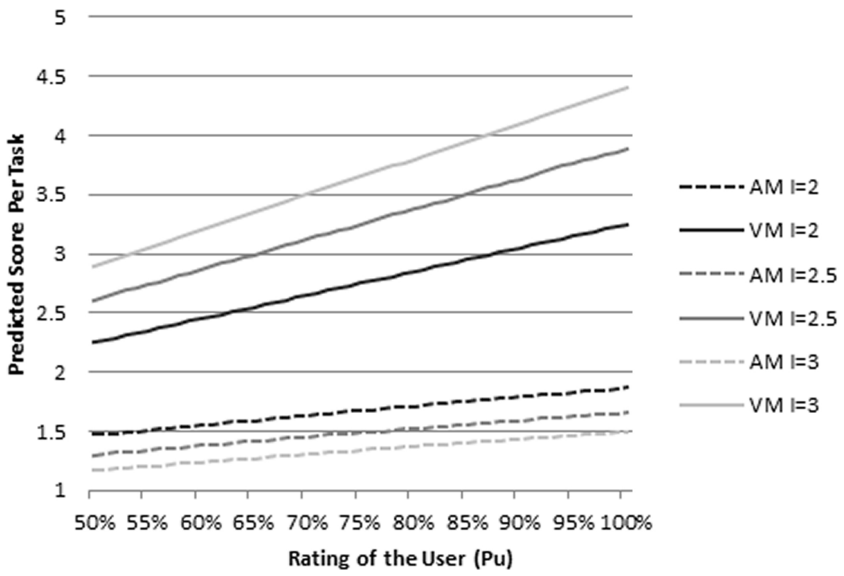


Figure 3.5: Simulation of score per task for different worker ratings, comparing Annotation Mode (AM) and Validation Mode (VM) with different interpretations (I) per task (P_{ub}=0.75).

3.2.5 Simulating the AV Model

The AV Model is simulated to predict a score per task (S) for a worker of a given rating (P_u) with the hypothesis that better workers will score more and hence be motivated to provide high-quality answers. For all the simulations there were eight annotating workers per task (U_A=8) and four validating workers per interpretation (U_V=4).[1]

Task difficulty The difficulty of the dataset will have an impact on the number of interpretations (I) that are submitted by the workers, with more difficult tasks having more interpretations. The score per task in Annotation Mode does not seem to be affected by the difficulty of the dataset, with highly rated workers only scoring slightly more. The score per task in Validation Mode is different between levels of difficulty, with harder tasks scoring more for higher rated workers (see Figure 3.5).

Quality of the crowd A measure of how well the workers (or user base) of the system are performing on average (P_{ub}) is essential when using a validation method.

[1]The model was simulated with eight annotators and four validators because this is the configuration of the *Phrase Detectives* system described in Chapter 4.

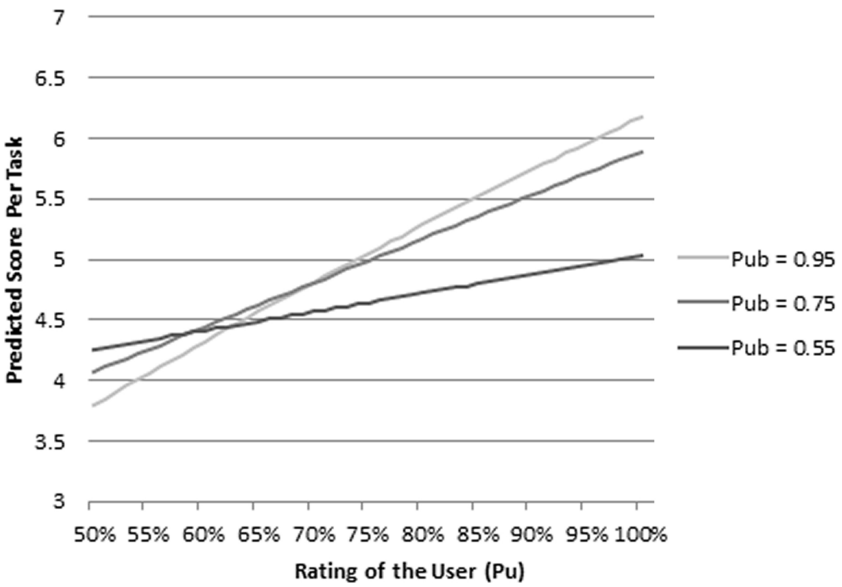


Figure 3.6: Simulation of score per task for different worker ratings, comparing different ratings (P_{ub}) for the user base (I=3).

The system increases the score of an annotation using validations so if the workers that are validating are not performing well this could have a negative impact, not only on the data quality, but also on the motivation of the workers. In three different scenarios of user base rating (P_{ub}=55% as near chance; P_{ub}=75% as an average response; and P_{ub}=95% as a good response) the model performs correctly, i.e. highly rated workers score more per task than poorly rated workers (see Figure 3.6). This effect is magnified when the workers are, overall, very good, but the model still rewards appropriately even when the workers are performing badly (close to chance).

Data maturity During the lifecycle of data being annotated with the model the worker will be presented with different proportions of annotation tasks compared to validation tasks (ϵ). When the data are initially released the worker will be given annotation tasks (ϵ=1). As more annotations are collected the number of validations presented to the worker increases until all tasks have been sufficiently annotated and only require validations (ϵ=0).

Higher-rated workers will score more per task and this increases as more validations are required (see Figure 3.7). This is due to higher-rated workers' annotations being

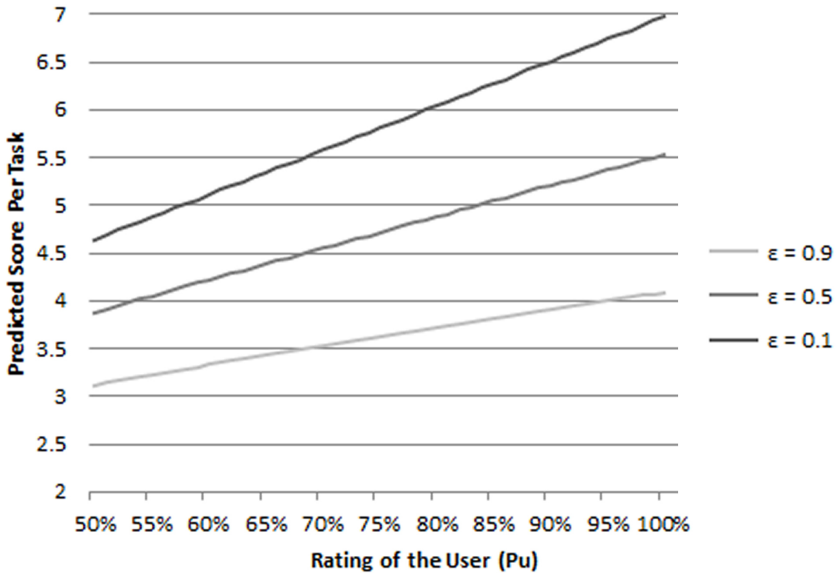


Figure 3.7: Simulation of score per task for different worker ratings at different stages of data maturity (I=3 and P_{ub}=0.75).

agreed upon more by validators and thus should increase the motivation of workers as the data matures.

The simulation of the AV Model shows that theoretically workers can be rewarded appropriately using retrospective agreement for tasks in which the solution is not known and workers should be motivated to provide higher quality solutions to increase their reward.

3.3 Social networks as AV Model systems

As previously discussed in Section 2.4.4 social networks can be used as a platform to increase exposure to the task, increase participation and perhaps improve quality. However, the social networks themselves can be viewed as an AV Model crowdsourcing system, combining features common to cQA systems in a more complex and sophisticated way that appeals to inherent, personal and social human motivations.

From simple requests ('Help me find my dog, please share') to more complex requests ('Does anybody know what this marine species is?'), social network users can create tasks and receive an answer in a very short space of time, either from annotation (another user replies with an answer) or a validation (other workers 'like' an

answer). Collectively these tasks could be viewed as a crowdsourcing approach using asynchronous validation. Firstly, this approach to crowdsourcing is defined, then it is tested against crowdsourcing criteria to see whether the assumption holds true.

3.4 Groupsourcing: A definition

In a similar way that crowdsourcing is defined as taking a job traditionally performed by a designated employee and outsourcing it to an undefined large group of Internet users through an open call [Howe, 2008], tasks can be completed by groups of workers of social networking websites that are self-organised and decentralised. The tasks are created by the workers, so they are intrinsically motivated to participate. The social nature of the groups allow workers to connect with others of similar interests, with the reward being able to have their problem solved or to benefit from the problem being solved. Social media are entertaining and the natural language of the interface allows users to express their emotions, appreciation, frustration, etc. The combination of these motivations that relate directly to motivations of crowdsourcing generally (see Section 2.3.1) may explain why this approach has evolved from the workers themselves.

Thus, a definition for **groupsourcing** is proposed as *completing a task using a group of intrinsically-motivated people of varying expertise connected through a social network* [Chamberlain, 2014b].

This is a more general definition than has been proposed before in relation to crowdsourcing disaster relief efforts [Gao et al., 2011] and could be applied to other cQA and opinion collection systems such as YahooAnswers[1], StackOverflow[2] and OpinionSpace [Faridani et al., 2010]. It is also a different definition from the term used to describe crowdsourcing team competition designs [Rokicki, Zerr, and Siersdorfer, 2015].

Groupsourcing combines three central principles of crowdsourcing (crowd wisdom, creation and voting) [Howe, 2008] and incorporates concepts of groupworking and group dynamics found in social psychology research [Forsyth, 2005]. The approach is also similar to crowd-powered websites such as iStockphoto[3] or Threadless[4], in which the creation and validation of content and metadata is managed by the users.

[1] `https://uk.answers.yahoo.com`
[2] `http://www.stackoverflow.com`
[3] `http://www.istockphoto.com`
[4] `https://www.threadless.com`

Table 3.5: A table of criteria that qualify the groupsourcing approach as crowdsourcing.

Crowdsourcing criteria	Groupsourcing
There is a clearly defined crowd.	The crowd is defined as a group on a social network.
There exists a task with a clear goal.	The task and goal are defined within a thread posted to the group.
The recompense received by the crowd is clear.	The group members socially learn about the topic they are interested in and gain peer recognition for their effort.
The crowdsourcer is clearly identified.	The crowdsourcer is the group member posting the task and their profile and interactions are visible to the group.
The compensation to be received by the crowdsourcer is clearly defined.	The member posting a task receives advice (and a set of solutions).
It is an online assigned process of participative type.	Group members actively participate in the process and may also be assigned to a particular task by another member.
It uses an open call of variable extent.	All group members may view and contribute to the task.
It uses the Internet.	Social networks are based on the Internet.

Is groupsourcing a type of crowdsourcing? Groupsourcing is distinguished by several features: data and tasks are created by the users; input is unconstrained and developed in series whilst simultaneously validated by the users themselves; users are inherently-motivated, socially-trained and work collaboratively; and the output is immediately accessible and beneficial to all, with users receiving recognition for their efforts (see Section 3.1).

As can be seen in the related work, crowdsourcing can come in many forms. An overarching survey of all prominent papers in crowdsourcing attempted not only to define what crowdsourcing means in terms of a definition, but also to define criteria in order to test if an approach is indeed what is considered to be crowdsourcing [Estellés-Arolas and González-Ladrón-De-Guevara, 2012]. Each criterion is explained and compared to the groupsourcing approach in Table 3.5 and it shows that groupsourcing

could be classified as a crowdsourcing approach.

3.5 Summary

This chapter has described features common to several approaches to harnessing the collective intelligence of crowds and outlines a model that uses a crowd not only to annotate data but also to validate those annotations, called the Annotation Validation (AV) Model. Simulation of the model shows that a validation step can be used to incentivise high quality when there is no access to a gold standard to judge worker responses.

Furthermore, it has been shown that social computing on networks (defined here as *groupsourcing*) can be described in the same terms as other crowdsourcing approaches and offers favourable conditions for collecting high-quality contributions from an engaged and self-motivated community of users. Whilst this will not be a revelation to the social computing research community, describing social networks in terms of crowdsourcing is a novel contribution that allows this promising research area to be analysed from a data-centric view.

In Part II, experimental work is undertaken to investigate whether social networks can overcome some of the barriers that have limited traditional crowdsourcing approaches such as low user engagement and poor-quality contribution. In Chapter 4 the idea that social networks are beneficial to deploy a system on is tested and in Chapter 5 the AV Model that is inherent in social networks is investigated to see if it offers greater quality than an annotation-only model. Finally in Chapter 6 inherent problem solving is investigated to see if it exists on social networks and, if it does, what level of quality does the community produce.

PART II: Collective Intelligence on Social Networks

4

Phrase Detectives: Benefits of deployment on social networks

Crowdsourcing interfaces can be linked to social networking sites such as Facebook to achieve high visibility and to explore different ways users can collaborate to exploit this enormous human resource. The social-computing approach to problem solving looks to overcome issues of user recruitment and participation, but presents new challenges such as how to access the data and how users interact with the interface.

This chapter investigates whether a problem-solving system deployed on a social network can gather more answers of a higher quality than a standalone system. Social networks have large numbers of users so it is intuitive to believe that a system deployed there would benefit from increased exposure to a larger user base and therefore participation would increase, especially if the system was integrated into the social features. Additionally, social networks work hard to ensure their users are real people and not companies, groups or spam [Stringhini, Kruegel, and Vigna, 2010] so the chance of poor quality answers being submitted might be lower.

These issues are investigated using *Phrase Detectives*, an online game designed to collect annotations about human language, with one system deployed as a standalone system and another deployed on the social network Facebook. Firstly, the *Phrase Detectives* game-with-a-purpose methodology is described, including terminology specific to the system and details of the annotation scheme. A summary of the data that were

Portions of this chapter previously appeared in Chamberlain and O'Reilly [2014]; Chamberlain, Kruschwitz, and Poesio [2012]; Chamberlain [2014a]; Poesio et al. [2013].

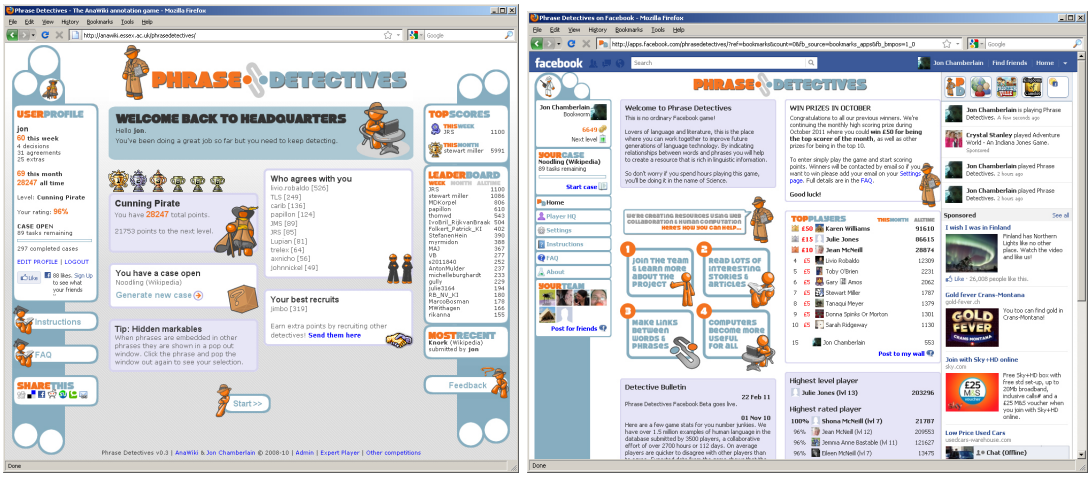


Figure 4.1: Screenshots of PD player homepage (left) and the PDFB homepage (right).

collected over six years and analysed is presented along with an analysis of player activity between systems. Finally, the contributions from the players are compared between systems, focusing on issues of data filtering and answer credibility.

4.1 Introduction

Phrase Detectives (PD)[1] is a text annotation GWAP designed to collect data about English anaphoric co-reference. The standalone version of the game was first released in December 2008. The **Facebook version of *Phrase Detectives* (PDFB)**[2], launched in February 2011, maintained the overall game architecture whilst incorporating a number of new features developed specifically for the social network platform (see Figure 4.1). Both interfaces were designed, developed and deployed by Jon Chamberlain as part of the *AnaWiki* project.

In most respects *Phrase Detectives* has all the features that would be anticipated from a GWAP (see Section 3.1): the data and tasks are managed by the administration; the player input is constrained and entered in parallel; players are mainly motivated by entertainment and social competitiveness, are explicitly trained and work collectively together; and the output is not of direct benefit to the majority of players. *Phrase*

[1] `https://www.phrasedetectives.com`
[2] `https://apps.facebook.com/phrasedetectives`

Detectives is different from other GWAPs in that the players validate the annotations, as well as enter the annotations themselves, and indirect financial incentives were extensively implemented over a long period of time.

4.2 Definitions

The description of the game uses terminology common in Natural Language Processing, but may be ambiguous with terms used in other domains. For that reason a selection of terms are defined here.

A collection of text documents is referred to as a **corpus** (plural, corpora) and are organised either by the source of the document, the primary language of the text, the theme of the documents, or all three. A document is divided into paragraphs and sentences, with smaller sections of text within sentences (typically noun phrases) referred to as **markables**.

The task in this study is anaphoric coreference, in which markables can be **an anaphor** of a previously mentioned named entity **antecedent** in the text (see Section 2.1). The interface collects two types of response from users, either an **annotation**, when the user chooses an appropriate selection of markables as a solution, or **validation**, when the user is asked to agree or disagree with a solution provided by another user. An annotation or validation decision from a user is described as a unit of **work**. A unique solution to the task is referred to as an **interpretation**, of which a task may have several before data collection is considered complete.

4.3 Data

The *Phrase Detectives* project was designed to collect annotations on novel corpora such as dictionary articles and narrative texts, rather than news articles that are more commonly available.

The texts come from two main domains:

- Wikipedia articles selected from the 'Featured Articles' page[1] and the page of 'Unusual Articles'[2];

[1] `http://en.wikipedia.org/wiki/Wikipedia:Featured_articles`
[2] `http://en.wikipedia.org/wiki/Wikipedia:Unusual_articles`

- narrative text from Project Gutenberg[1] including a number of short stories (e.g. *Aesop's Fables*, Grimm's Fairy Tales, Beatrix Potter's tales) and more complex narratives such as several Sherlock Holmes stories by A. Conan-Doyle, *Alice in Wonderland*, and several stories by Charles Dickens.

The corpus contains 806 documents, totalling 1,185,911 words (see Table 4.1).

4.4 Annotation scheme

The corpus was annotated according to the linguistically-oriented approach to anaphoric annotation that is currently prevalent, having been adopted in OntoNotes [Pradhan et al., 2007], the ARRAU corpus [Poesio and Artstein, 2008] and in all the corpora used in the 2010 SEMEVAL anaphora evaluation [Recasens et al., 2010]. In this type of annotation, all noun phrases (NP) are considered markables, and anaphoric relations between all types of entities are annotated (for example coordinated NPs such as 'John and Mary' which also considered markables) unlike the practice in the MUC and ACE corpora[2].

Players can assign four types of interpretation to markables:

- DN (discourse-new): this markable refers to a newly introduced entity;
- DO (discourse-old): this markable refers to an entity already mentioned (the user must specify the closest mention by character distance);
- NR (non-referring): this markable is non-referring (e.g. the pleonastic *it* in 'It is raining');
- PR (property): this markable represents a property of a previously mentioned entity (e.g. *a teacher* in 'He is a teacher').

4.5 Methodology

The game uses two styles of text annotation for players to complete a linguistic task. Initially text is presented in **Annotation Mode** (called Name the Culprit in the game,

[1] `http://www.gutenberg.org`
[2] `http://projects.ldc.upenn.edu/ace/data`

see Figure 4.2). This is a straight-forward annotation mode in which the player makes an annotation decision about a highlighted markable. If different players enter different interpretations for a markable then each interpretation is presented to more players in **Validation Mode** (called Detectives Conference in the game, see Figure 4.3). The players in Validation Mode have to agree or disagree with the interpretation.

Player workload is organised around a case: a block of text from a document in which a certain number of markables have been allocated as tasks in either Annotation or Validation Mode. The tasks in a case are presented to the player in order of appearance in the text. The algorithm for generating new cases aims to maximise variety (i.e. making sure that players rarely see the same text twice) over completion rate (i.e. maximising the rate at which documents are completed).

The fundamental elements of *Phrase Detectives* are apparent in both versions of the game. The technical details of the implementation of both systems is outlined in Appendix C.

4.5.1 Game design

The realisation of the detective metaphor in the game's graphical design is achieved in part through graphical devices (e.g. the buttons are stylised with a cartoon detective character) and in part through the text on the pages, written as if the player was a detective solving cases. The detective metaphor is also reflected in the level system used in the game to foster the experience of progression. Players begin at the rookie level and then achieve progressively higher detective-related levels.

PDFB includes many refinements and bug fixes, including cleaner imagery and faster overall gaming experience by removing the scoring feedback screen. Data generated from this version of the game are compatible with previous versions and both current implementations of the game run simultaneously on the same corpus of documents.

Annotation Mode (Name the Culprit) Annotation Mode is the primary activity dedicated to the labelling of data by players. The players are shown a window of text in which a markable is highlighted in orange, as shown in Figure 4.2.

Moving the cursor over the text reveals the markables within a bordered box. To select a markable the player clicks on the bordered box and the markable becomes

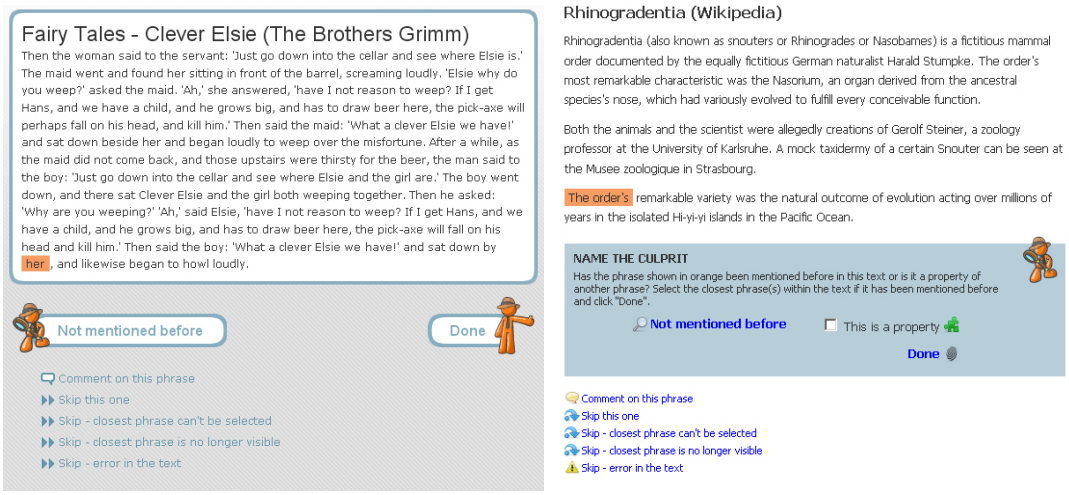


Figure 4.2: Screenshots of Annotation Mode in PD (left) and PDFB (right)

highlighted in blue. This process can be repeated if there is more than one antecedent (e.g. for plural or coordinated anaphors such as *'they'*). When the player has made their selection according to the annotation scheme in Section 4.4 the annotation is submitted by clicking the `Done!` button.

The choice among candidate antecedents is carried out with respect to a context window, the portion of text displayed to the player. Specifying the anaphoric interpretation of markables crucially depends on being able to point to the last mention of an entity in a context, yet to avoid scrolling players cannot be presented with too much context. The distance between entity mentions suggests that, for anaphoric expressions, the majority of entities not mentioned in the current or previous sentence [Hitzeman and Poesio, 1998; Hobbs, 1978] are mentioned in four sentences or fewer [Vieira and Poesio, 2000]. Therefore, the context window was set to be at least 1,000 characters, rounded up to the nearest sentence so as to fit comfortably within a single browser page at a standard 1024x768 resolution. The context ends with the sentence which contains the highlighted markable and markables after the highlighted markable cannot be selected so as to present a uni-directional reading task to the player.

Each markable in a case is presented to several players in Annotation Mode.[1] If every player chooses the same interpretation (for example, they all say the entity is

[1]By default each markable is presented eight times in Annotation Mode, see Section 3.2.1 for justification.

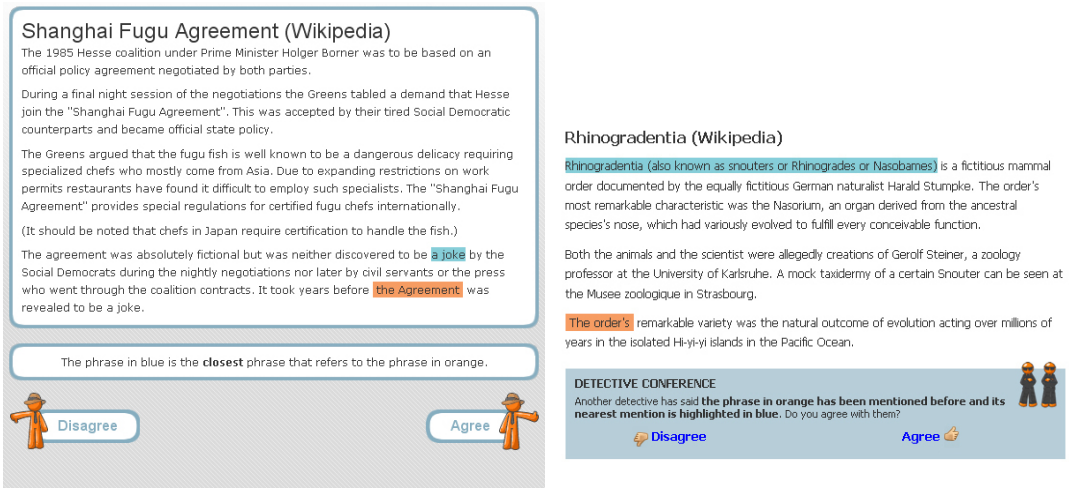


Figure 4.3: Screenshots of Validation Mode in PD (left) and PDFB (right)

Discourse New, i.e. it has not been mentioned before) then that markable is classified as complete. Otherwise, it is entered among the markables to be validated through Validation Mode (Detectives Conference), discussed next.

Given that players are only allowed to choose between a limited range of options (e.g. they are not allowed to mark bridging interpretations or discourse deixis[1]) and there are restrictions on the context window, the players are also allowed to skip tasks and/or submit a comment about markables.

Validation Mode (Detectives Conference) Every markable for which multiple interpretations have been proposed in Annotation Mode must go through Validation Mode (called Detectives Conference, see Figure 4.3). Both the candidate markable and the antecedent markables are highlighted, in orange and blue respectively. If the player disagrees with the proposed interpretation they enter Annotation Mode in order to enter an alternative interpretation. If the interpretation they specify has not been entered before this will also be entered into the Validation Mode (see Figure 3.4 for a diagrammatic representation of this system). Apart from making the game more interesting, it was assumed that validating annotations would be faster than creating annotations [Chklovski and Gil, 2005].

[1]See Appendix E for examples.

4.5.2 Training and evaluating players

The tasks in the game require a complexity of judgements from the players. Yet clearly it cannot be expected that players be experts about anaphora, or be willing to read a manual explaining how anaphora works, so the majority of training has to be done while playing the game.

After reading brief instructions of how to play the game, the main training mechanism is by explicitly asking players to annotate text which has already been annotated by an expert (gold standard text) and their level of understanding can also be assessed. Contextual help information about the task is also presented to the players during the game.

Players always receive a training text when they first start the game. The training texts show the player whether their decision agrees with the gold standard (unambiguous markables are used in these cases, to avoid confusion). Once the player has completed all of the training tasks they are given a **user rating** (the percentage of correct decisions out of the total number of training tasks). The user rating is recorded with every future annotation or validation decision. Players are given training texts until the rating is sufficiently high enough to be given real text from the corpus.[1] The training tasks also prevent automated form-completion software and malicious players from progressing far in the game.

In PDFB a training document must be completed at every level of promotion and the game asks the player to keep doing training documents until the rating threshold is achieved. The rating threshold is increased at higher levels. PDFB also allows players to do a training document whenever they want, called 'Head-to-Head' mode in the game. This feature was particularly useful for players who were interested in the game, but English was not their native language, from informal sessions with ESL (English as a Second Language) students at the University of Essex.

Players learn about correct decisions by reinforcement through Validation Mode. This builds on the assumption that the majority of players will agree with a good decision, which is not always the case especially if the markable is complex or ambiguous. However, generally speaking, scoring high points in Validation Mode is an indication

[1] A minimum rating threshold of 50% is set for the game, see Section 3.2.1 for justification.

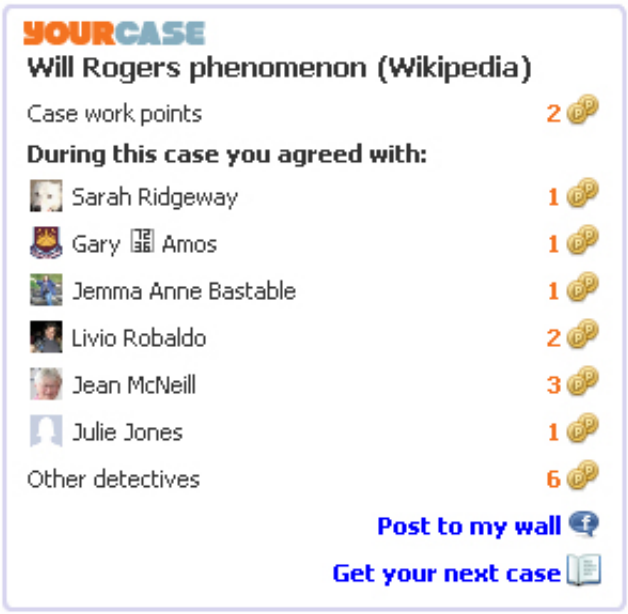


Figure 4.4: Detail of the reward screen in PDFB, displayed at the end of each case, showing the player how many points they scored and who they agreed with.

of a good interpretation (see Section 3.2.5 for a simulation of the scoring system under different conditions).

4.5.3 Motivating players

Scoring points and game progression Scoring points is one of the most important incentives in the game. Through scores, players gain a sense of achievement and compete with other players.

During training, the main function of scoring is to teach players about anaphora by comparing their judgements with those in a gold standard. This goal can be achieved simply by having players score points by assigning to a given markable the same interpretation that can be found in the gold standard.

When players go past the training level, the way their points are counted in the game changes. The goal now is to motivate them to think carefully about what they do. In order to do this, the scoring mechanism was designed so that players can get more points when other players agree with them than they would by randomly choosing interpretations.

Figure 4.5: Detail showing criteria for the next level in PDFB, displayed to the player on their homepage.

In Annotation Mode, trained players get one point every time they produce a judgement, to encourage them to engage in this activity. In addition, players producing a judgement in Annotation Mode get an extra point for that judgement every time another player agrees with it in Validation Mode. If only one interpretation for a markable is chosen by all players being presented that particular markable in Annotation Mode, then all of these players get awarded an extra agreement point, but that interpretation is not presented in Validation Mode.

Players in Validation Mode who agree with an interpretation get one point for every player who entered that interpretation in Annotation Mode. If they disagree with it, they get one point for every player who entered another interpretation while in Annotation Mode. They are also asked to propose an alternative interpretation for that markable. Only the initial annotating players gain points from retrospective agreement; further players gain their points from Validation Mode. This is an implementation of the AV Model discussed in Section 3.2.3.

The scoring system was also designed to provide an incentive for players to return and inspect the scoreboard as they gain points retrospectively. After scoring a certain number of points the player is promoted to the next level. Lower levels require fewer points in order to encourage new players to keep playing, but progressing to a higher level gets increasingly harder.

Scores in PDFB are added at the end of a case, rather than after each task in PD, which encourages completion of all the tasks allocated (see Figure 4.4). After each task

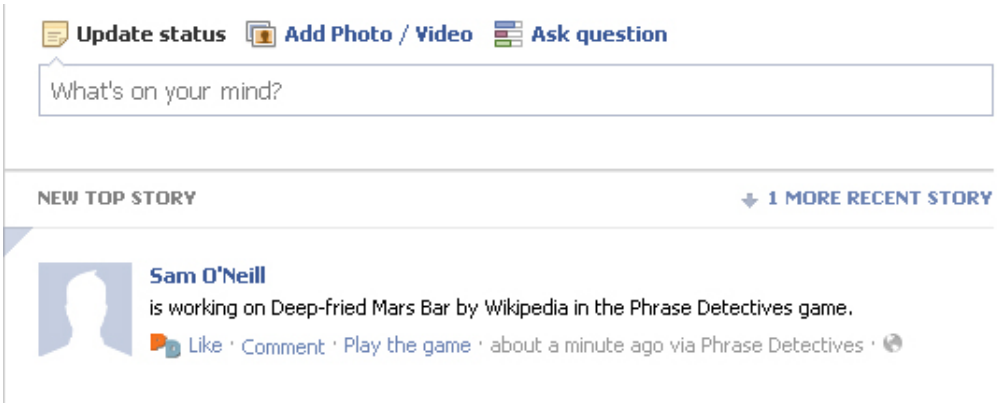


Figure 4.6: Detail of a news (or wall) post created automatically from the game, as seen by the player's friend.

feedback on the player's decision is presented in the left-hand menu as a phrase such as 'Perfect!' or 'Good agreement!' depending on how many other players agree with the decision. Player levels have well-defined criteria and the player must activate the new level once the criteria are met (see Figure 4.5).

The game features incentives usually found in online games for players motivated by a competitive spirit, such as weekly, monthly and all-time leaderboards, cups for monthly top scores and named levels for reaching a certain number of points. In addition to leaderboards visible to all players, each player can also see a leaderboard of the players who agreed with them the most. Although this leaderboard provides no direct incentive (as you cannot influence your own agreement leaderboard) this feature reinforces the social aspect of the scoring system. PDFB also has leaderboards for the highest level players, highest rated players and the players with the biggest team.

Incentives on social networks PDFB has additional features designed to take advantage of the social nature of social networks. News feed (or wall) posting is integrated into the PDFB game. This allows a player to make an automatically generated post to their news feed which will be seen by all of the player's friends (see Figure 4.6).[1]

The posts include a link back to the game. Players are required to make a post from the game every time they are promoted to the next level. Posting is a very important

[1]Since the release of PDFB Facebook has changed how posts are displayed. Posts from PDFB now appear on the player's Facebook profile and in a news ticker.

factor in recruiting more players as studies have shown that the majority of social game players start to play because of a friend recommendation.[1]

Any of the player's friends who are playing the game form the player's team, which is visible in the left hand menu. Whenever a player's decision agrees with a team member the player scores double points, which is highlighted on the reward screen.

User experience and game control The choice of documents was considered important in getting players to enjoy the game, to understand the tasks and to keep them playing. Whilst some of the chosen texts were straightforward, others could provide a serious challenge to readers, in particular when the task is resolving anaphora. Texts were manually graded by administrators[2] for complexity (on a scale of one to four) after import. Players could choose the maximum level of document complexity they wish to read as they may be motivated to play the game to improve their English skills, or equally because they enjoy reading challenging texts. Players could also specify a preference for particular topics.

Timing constraints are a key aspect of what makes games exciting [von Ahn and Dabbish, 2008], but in this game there were no timing constraints. This decision was based on the results of the first usability study of PD, discussed in Section 4.5.4. In the game prototype, players could see how long they had taken to do an annotation. In contrast with the idea that timing provides an incentive, the players complained that they felt under pressure and that they did not have enough time to check their answers, even though the time had no influence on the scoring. As a result, in all following versions of the game the time it takes players to perform a task is recorded but not shown.

A player who has a profile in both versions of the game could create a link between them on the Settings page. This transfers the players' settings to the Facebook version of the game, as well as the record of which documents they have completed so they are not asked to do them again. This link allows a comparison of how the same user performs on the two different platforms.

[1] `http://www.lightspeedresearch.com/press-releases/it's-game-on-for-facebook-users`
[2] Jon Chamberlain manually graded the English documents.

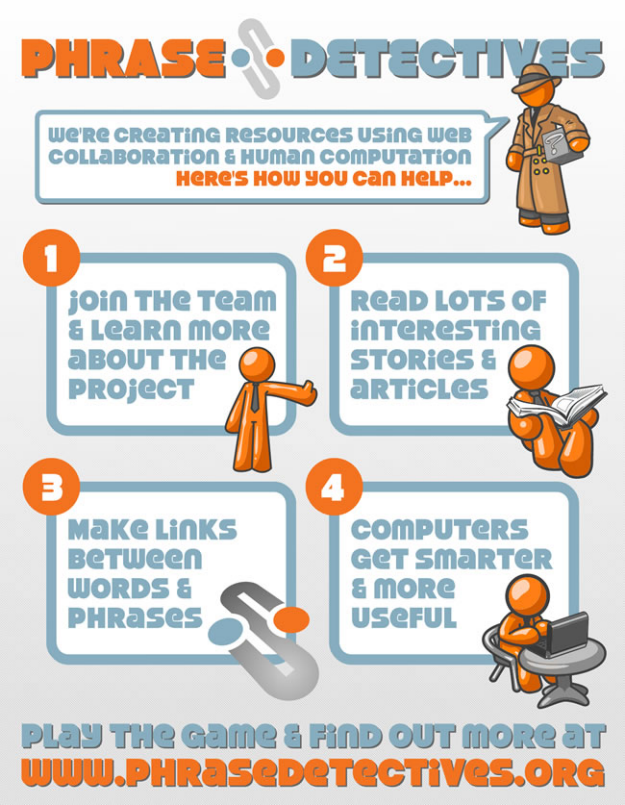


Figure 4.7: Postcard used for promoting *Phrase Detectives*.

Indirect financial incentives Monthly prizes for the highest-scoring players in the form of Amazon[1] shopping vouchers sent by email were offered regularly. The monthly prize motivates the high-scoring players to compete with each other by doing more work, but also motivates some of the low-scoring players in the early parts of the month when the high score is low. Prizes were also awarded by randomly selecting an annotation. These prizes motivate low-scoring players because any annotation made during the prize time period has a chance of winning (much like a lottery) and the more annotations you make, the higher your chance of winning. These prizes were sometimes awarded as an alternative to the highest-scoring prizes and sometimes in addition to those prizes.

The prizes have ranged from £5-10 ($7.50-15) daily, £10-15 weekly ($15-22.50), and from £30 ($45) to £75 ($132.50) for the monthly high scoring prizes. Full details of

[1] `http://www.amazon.co.uk`

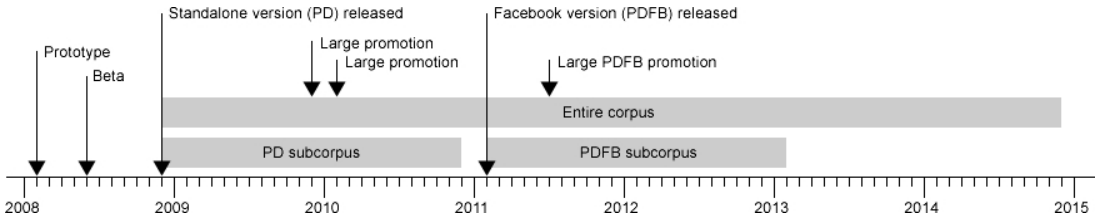


Figure 4.8: Timeline of the release of the two interfaces of *Phrase Detectives*.

the prizes allocated each month are in Appendix B.

4.5.4 Usability testing with a prototype

A prototype of the game was built to test initial ideas about the game format and task design, using a small corpus of Aesop fables. This prototype was tested in February 2008 with a group of 16 players (staff and students at the University of Essex) who were paid a small amount (£10) to play the game for an hour whilst their actions and verbal feedback were recorded. This study led to interface refinements, in particular reducing task feedback (why the points were scored and how long it took to complete the task) and removing timing constraints, as well as better instructions and examples of the tasks. A beta release of the game to friends and the linguistic community took place in June 2008 to identify and fix bugs [Chamberlain, Poesio, and Kruschwitz, 2008].

4.5.5 Promotion

The campaign to attract the general public began with press-releases in January 2009 that were picked up by *Science Daily*[1] and *Innovations Report*[2], among other online publications, and by *Times Higher Education* among the regular academic journals, and Jon Chamberlain was interviewed by the BBC Radio. In addition the game was written about on blogs such as *Computer Science for Fun*[3] and was listed on bookmarking websites and gaming forums.[4] A pay-per-click advertising campaign was used on the social networking website Facebook.

[1] http://www.sciencedaily.com/releases/2009/01/090126082345.htm

[2] http://www.innovations-report.com/html/reports/information-technology/networked-human-computation-solve-computer-language-126034.html

[3] http://www.cs4fn.org/linguistics/phrasedetectives.php

[4] http://www.gamescanteach.com/category/games/phrase-detectives

Table 4.1: *Phrase Detectives* corpus summary at 30 Nov 2014.

	Completed			Total		
	Docs	Words	Markables	Docs	Words	Markables
GNOME	5	875	275	5	875	275
Wikipedia	379	183,023	57,338	591	823,768	267,638
Gutenberg	139	117,314	37,413	208	355,143	115,079
User	1	1,012	389	2	6,125	2,223
	524	**302,224**	**95,415**	**806**	**1,185,911**	**385,215**

Table 4.2: Players of *Phrase Detectives* as of 30 Nov 2014.

	PD	PDFB	Linked
Total players	37,525	1,069	40
Total players with a rating	2,466	280	40
Proportion of players with a rating	6.6%	26.2%	100%

Efforts to reach out to the Computational Linguistics community in the first year involved announcements through mailing lists such as the Linguist List and Elsnet, as well as presenting the game in a number of seminars, workshops, and conferences. Postcard-size flyers (see Figure 4.7) were also distributed. The efforts to reach out to this community intensified during the first recruitment campaign of January 2010 during which the game was mentioned on blogs such as *Language Log*.[1]

4.6 System summary and datasets

Since the first release of the game on 1 December 2008 to 30 November 2014 (six years) 524 documents have been fully annotated, for a total completed corpus of 302,224 words and 95,415 markables, 25% of the total size of the collection currently uploaded for annotation in the game (1.2M words, see Table 4.1).

The size of the completed corpus does not properly reflect the amount of data that have been collected, as the case allocation strategy adopted in the game privileges variety over completion rate. As a result, all the 806 documents in the corpus have

[1] `http://languagelog.ldc.upenn.edu/nll/?p=2050`

Table 4.3: Total responses of the two modes in the two interfaces of *Phrase Detectives* as of 30 Nov 2014.

	PD	PDFB	Total
Annotations	1,296,518	705,788	2,002,306
Validations (Agree)	176,416	238,354	414,770
Validations (Disagree)	358,372	247,740	606,112
	1,831,306	1,191,882	3,023,188

Table 4.4: Summary of datasets analysed in the results section.

Dataset	Description	Docs	Words	Anns	Vals	Players
Full corpus	01 Dec 08 - 30 Nov 14	524	302.2k	2.0M	1.2M	38.6k
PD	01 Dec 08 - 30 Nov 10			1.1M	394.0k	
PDFB	01 Feb 11 - 31 Jan 13			506.6k	309.0k	

been partially annotated and it is estimated that the corpus is in fact 35% complete.[1]

38,594 players have registered, 2,746 of whom went beyond the initial training phase (see Table 4.2). These players did more than 5,000 hours of work, i.e. 2.5 person-years and produced over three million annotations and validations (see Table 4.3).

The dataset from the first six years was reduced to two smaller datasets for analysis, which represent the first two years of each interface being operational. The PD subset does not overlap with PDFB because the latter was not live. The PDFB subset was captured at a time that, whilst overlapping with PD, the focus was not on that interface and there was comparatively little activity (see Table 4.4 and Figure 4.8).

Statistical analysis The data were analysed by exporting from the MySQL database using PHP, then converting to XLS spreadsheets or CSV files to be analysed in R.

Paired and unpaired t-tests were used for comparing datasets when the distribution was anticipated to be normal. When unpaired data were expected to be non-parametric (with a non-normal distribution) a Mann-Whitney U-test was used. Z-tests were used to compare population proportions. Chi square tests were used for categorical data.

[1]The completion estimation is based on an approximation algorithm using the number of annotations received and the number of validations required to complete the markable during the game process.

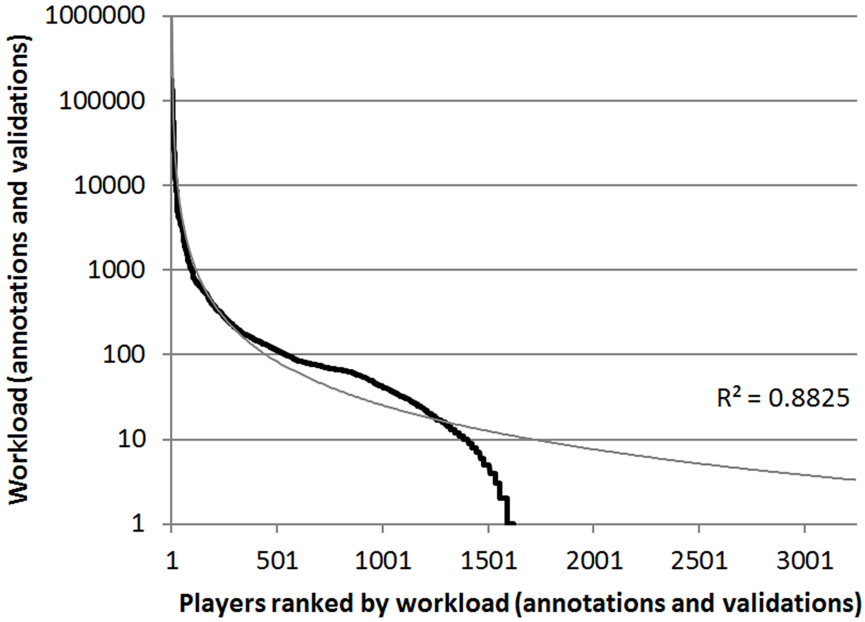


Figure 4.9: Figure showing PD players ranked by workload (annotations and validations) and a Zipf power curve.

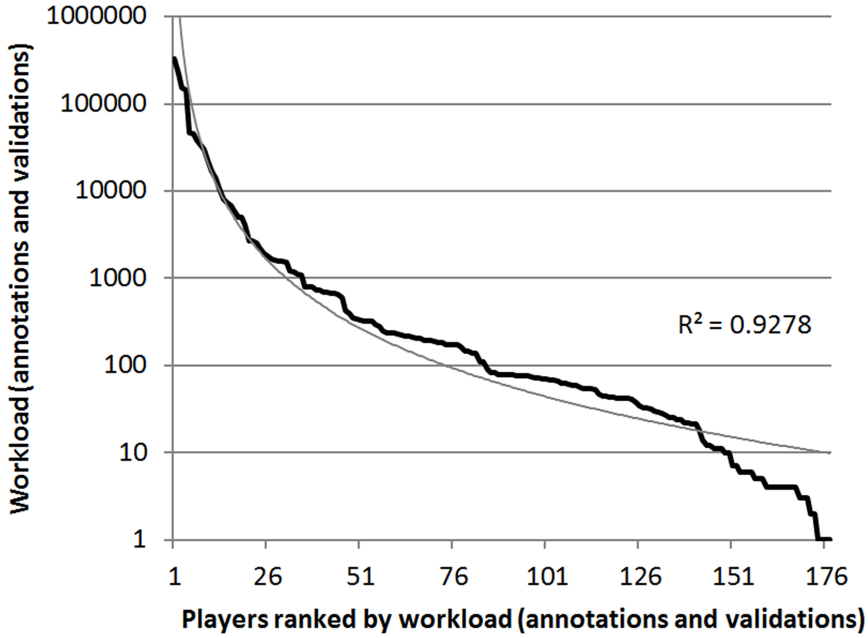


Figure 4.10: Figure showing PDFB players ranked by workload (annotations and validations) and a Zipf power curve.

Table 4.5: Closeness of fit of the Zipf power curve for workload of players of PD and PDFB under different filtering conditions.

	PD		PFBD	
Workload filter	**R^2**	**Data loss**	**R^2**	**Data loss**
None	0.883		0.928	
<= 10	0.975	0.07%	0.977	0.01%
<= 15	0.985	0.13%	0.982	0.02%
<= 20	0.991	0.20%	0.982	0.02%
<= 50	0.993	0.73%	0.982	0.09%

Pearson's correlation was used to test the relationship between variables when it was expected to be linear, Spearman's rank correlation coefficient was used when the relationship was expected to be non-linear and in some cases both coefficients were reported when the relationship was unknown. P values are reported unless they have an alpha level of $p<0.01$.

4.7 User activity

There is a considerable difference between the number of players registered for the two versions of the games (see Table 4.2); however, the most important consideration is whether the player will complete the training in order to provide useful annotations and validations.

PDFB has a much higher conversion rate (26.2%) of registered players to trained players than PD (6.6%) (PD n(37,525), PDFB n(1,069), $p<0.01$, z-test), most likely because the registration process of PDFB requires the player to be registered to Facebook and accept the game's permissions. This puts off casual users and those that commit to trying the game are more likely to continue through the training. To register for PD a user simply provides a username and password (in order to put as few obstacles in the way of registering); however, this has been subject to automated registrations from spambots. This does not pose a threat to the data integrity as the spambot cannot submit any real data without passing the training phase.

Additionally, there are 40 players who have created a link between their profiles on the two versions of the game. All linked players had completed training suggesting this

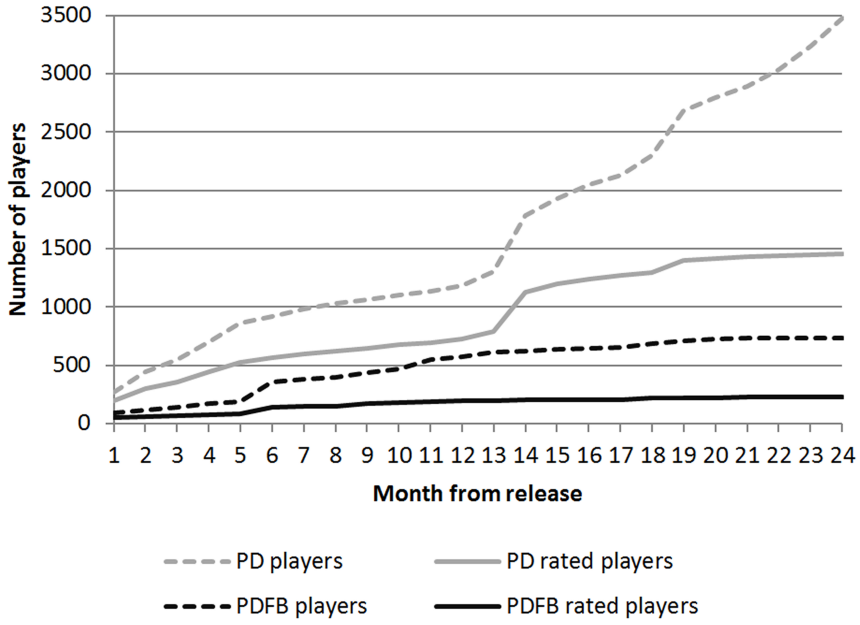


Figure 4.11: Chart showing total number of players and rated players for the first 24 months of release of both PD and PDFB.

is something more advanced players do.

4.7.1 Workload

The workload of players was investigated by ranking all players from the entire corpus by the amount of work (annotations and validations) they had completed. This is a more accurate measure of workload than completed tasks or score due to the system's design.

As expected (see Section 2.3.1) both systems' player workload follow a Zipf power distribution (R^2=0.883 for PD and R^2=0.928 for PDFB, see Figures 4.9 and 4.10). Both systems show deviation from the power curve after players have done approximately fewer than 50 annotations and validations. This is an indication that these are players who are still trying out the game but quickly give up.

The data were subsequently filtered for players with a low workload to see if the closeness of fit would be improved (see Table 4.5). By filtering players who have done 20 or fewer annotations and validations the closeness of fit to a power curve is 0.99 for PD and 0.98 for PDFB, with a negligible loss of data.

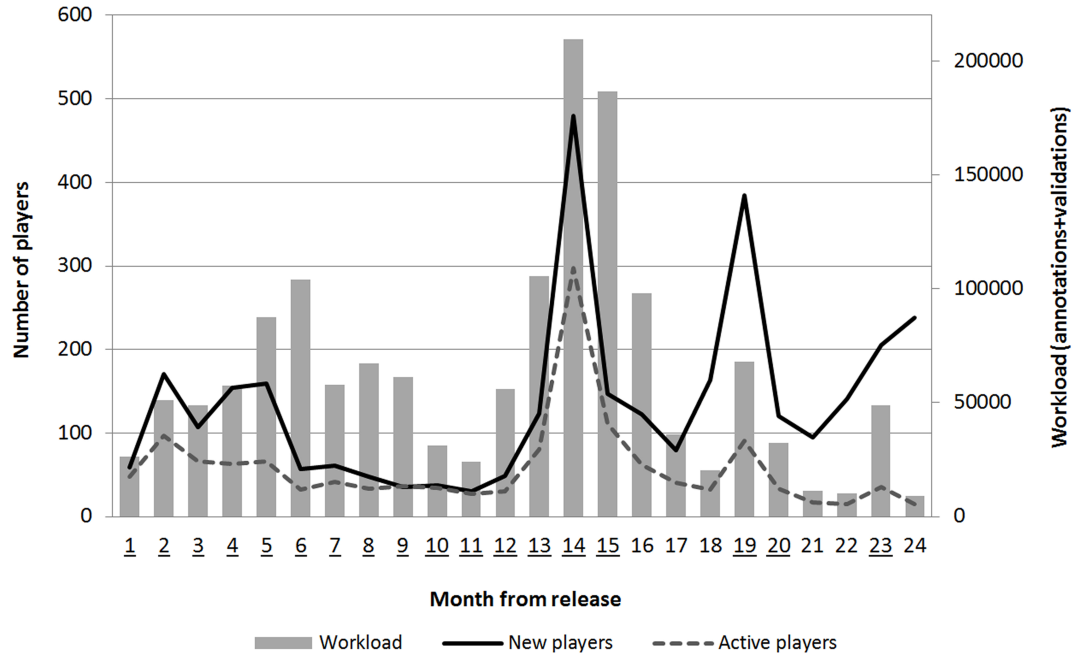


Figure 4.12: Chart showing work per month plotted with new players and active players in the first 24 months of release of PD (months with financial prizes are underlined).

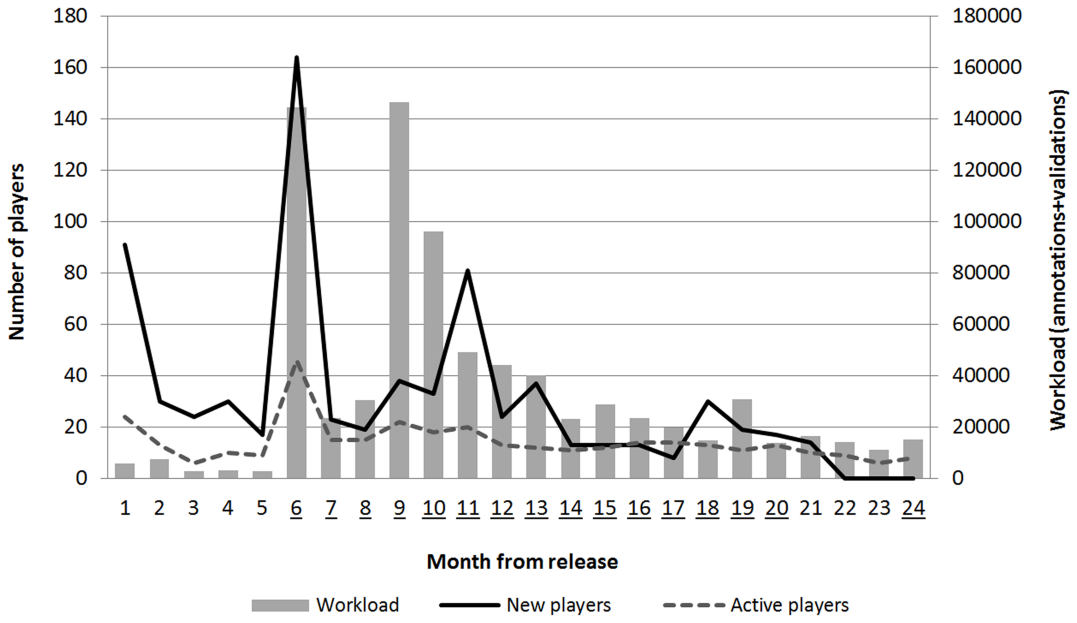


Figure 4.13: Chart showing work per month plotted with new players and active players in the first 24 months of release of PDFB (months with financial prizes are underlined).

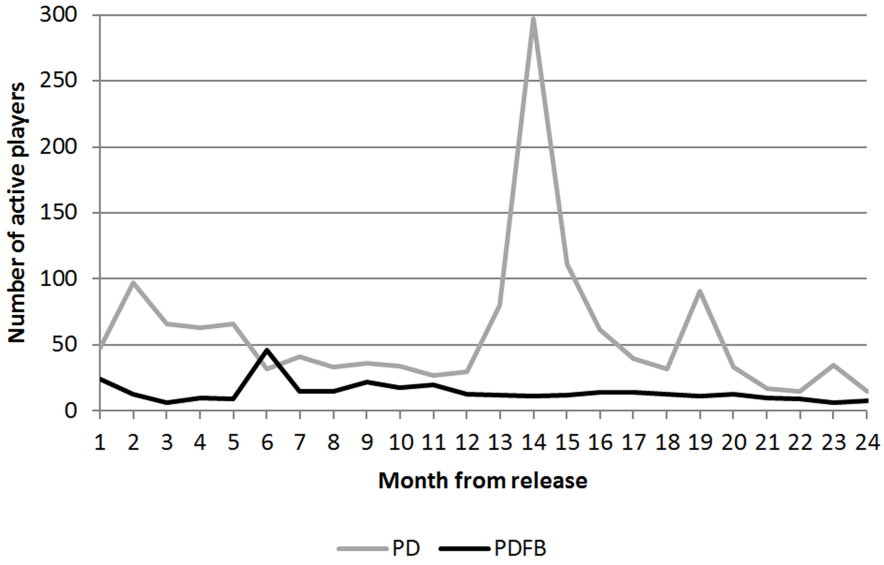


Figure 4.14: Chart showing the number of active players for the first 24 months of release of both PD and PDFB. PD had more active players than PDFB (58.4 sd(57.2) compared to 14.3 sd(8.1), $p<0.01$, paired t-test).

4.7.2 Recruitment and participation

Both versions of the game saw a steady increase in players over the first 24 months of release (see Figure 4.11) with jumps in recruitment when there were promotional efforts (see Table B.1 and Table B.2 in Appendix B). After 14 months of PD's release player recruitment appears to continue to rise more rapidly than PDFB; however, the conversion of these players to rated players shows that perhaps this rise was due to an increase in spam registrations.

Game activity was investigated to find out how many players were active each month, defined as whether a player made an annotation or validation, and how much work they do (see Figure 4.12 and 4.13). The number of active players tends to spike in months when there were promotional efforts and large financial rewards. On average PD had more active players than PDFB (58.4 sd(57.2) compared to 14.3 sd(8.1), $p<0.01$, paired t-test), see Figure 4.14.[1]

However, when looking at the workload of active players, PDFB players did more work than PD players (2,077 sd(1,535.6) compared to 1,167.2 sd(633.4), $p<0.01$, paired

[1] A paired t-test is used for statistical analysis using the month after release as the paired factor.

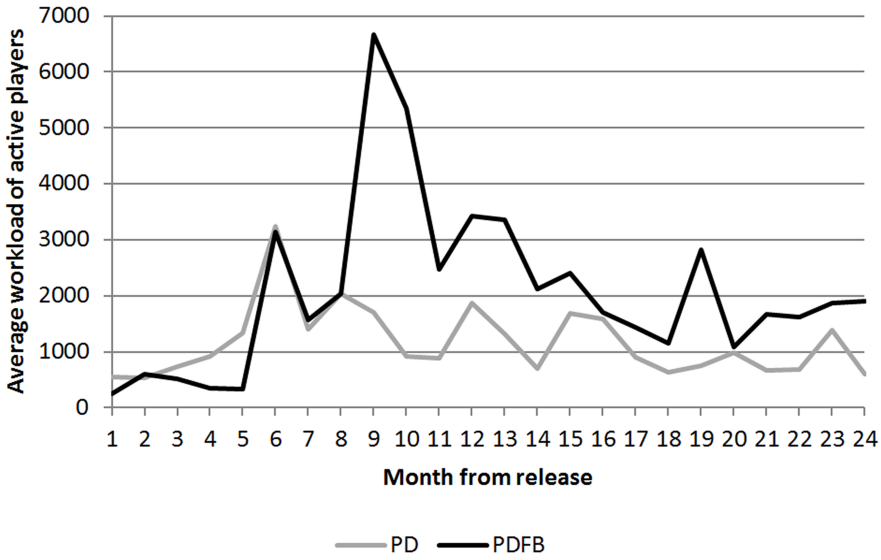


Figure 4.15: Chart showing the average workload of active players in the first 24 months of release of both PD and PDFB. PDFB active players did more work compared to PD active players (2,077 sd(1,535.6) compared to 1,167.2 sd(633.4), $p<0.01$, paired t-test).

t-test), see Figure 4.15.

The effectiveness of incentives was analysed by looking at new players, active players and new annotations each month. Table B.1 and Table B.2 in Appendix B show the recruitment and player activity of the first 24 months of release of the two games. They also show the months when financial incentives were offered in the form of top-scoring and lottery-style rewards and for these months a work per unit cost can be calculated. There was no difference between the two systems work gained per unit cost of prizes applied (PD 437.1 sd(486.6), PDFB 395.6 sd(320.7), $p=0.774$, unpaired t-test).

There is a strong positive correlation between the amount of prize funds on offer and the total work done by all players of both games (PD, n(24), $R=0.566$; PDFB n(24), $R=0.648$, Pearsons).[1] This implies that implementing financial incentives into a game will generate more work from the community. Analysing prize funds with work per month is problematic due to build-up effects in the month prior (when the prizes

[1]See Table B.1 and Table B.2 for a full breakdown of Pearson and Spearman correlation values. Both values are given as the relationship is hypothesised to be linear, in which case Pearson's correlation would be appropriate; however, potential skewing due to outliers suggests Spearman's rank coefficient would be a more robust test.

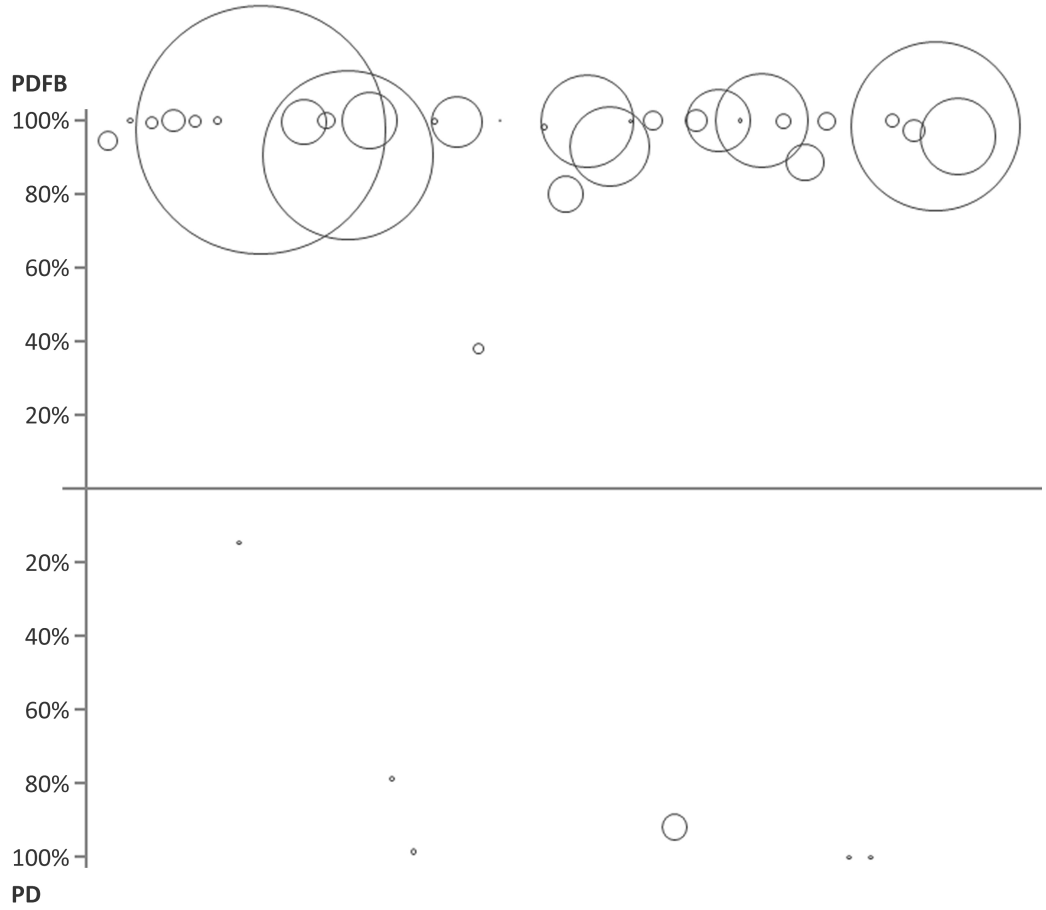


Figure 4.16: Bubble chart showing the proportional workload of linked players on the two interfaces after they had linked their accounts. Each bubble represents a player, the bubble area reflects the total amount of work (annotations and validations) of the player since linking and the vertical axis is the proportion of work done on each interface since linking.

are advertised to players) and residual effects in the following months when players continue playing because they are motivated by winning.

Whilst it could be expected that the number of active players steadily increases over time as more players are recruited, the results show that most players will play the game for a short period of time and only a small number continue to play every month.

4.7.3 Player preferences

In order to investigate which interface players preferred without explicitly asking them, the workload of the 40 players who had linked their accounts in both systems was analysed. These players on average did more work after linking their account (counting work on both systems); however, the difference was not significant due to low sample size and huge variations in the amount of work done (n=40; before mean 17,660.5 sd(42,593.2); after mean 22,412.2 sd(60,948.8); paired t-test).

After the accounts were linked the majority of players preferred to use PDFB exclusively. Most importantly these players did considerable amounts of work on the PDFB platform compared to those who preferred the PD platform (see Figure 4.16).

4.8 Quality of decisions made on social networks

In this section the quality of the players' decisions are compared between PD and PDFB to investigate whether deploying a system on a social network can result in higher-quality decisions (as compared to a gold standard, see Appendix D). In addition, several filtering mechanisms are tested to remove spam and outlier decisions to obtain a more accurate representation of player quality in each system and to see whether either system is particularly affected by large amounts of poor data.

Annotation and validation decisions are measured by precision (the proportion of decisions that are true gold standard interpretations), recall (the proportion of gold standard interpretations that were correctly identified), accuracy (the proportion of decisions that are correct compared to the gold standard) and F-measure (F1), the harmonic mean of precision and recall, used as an overall performance measure (see Table 4.6).

$$Precision = \frac{TP}{TP+FP}$$

Table 4.6: How precision and recall are calculated from player decisions.

	Gold standard positive	**Gold standard negative**
Player positive	TRUE POSITIVE (TP) Player annotation agrees with gold standard. Player makes an agreement validation with a gold standard.	FALSE POSITIVE (FP) (TYPE I ERROR) Player annotation is not the gold standard. Player makes an agreement validation with an interpretation that is not the gold standard.
Player negative	FALSE NEGATIVE (FN) (TYPE II ERROR) Player makes a disagreement validation on a gold standard interpretation.	TRUE NEGATIVE (TN) Player makes a disagreement validation on an interpretation that is not the gold standard.

$$Recall = \frac{TP}{TP+FN}$$

$$Accuracy(Agreement) = \frac{(TP+TN)}{N}$$

$$F1 = 2 * \frac{(Precision*Recall)}{(Precision+Recall)}$$

Baseline quality of annotation and validation decisions Each annotation and validation decision from the two interfaces on the Gutenberg (G1) and Wikipedia (W1) corpora were analysed to create a baseline level of quality. The results show that PD performs better on G1 and W1 (F=0.86/0.81) than PDFB (F=0.77/0.73). This could be a result of the PD interface being used earlier to annotate the markables, the implication being that the easier markables would be completed first leaving the more difficult and ambiguous markables to be annotated by both systems in validation mode. It may also be the result of outliers, spam and other causes of poor decisions (see Section 2.3.2).

4.8.1 Filtering to remove poor quality decisions

The goal of filtering is to identify and remove poor-quality decisions in order to increase the overall quality of the remaining decisions and a number of filters were tested here:

	System	Good player	Bad player
ANNOTATIONS			
Total Annotations:	1423078	4587	11018
Average Annotation Time:	00:00:07	00:00:07	00:00:04
Total (Ratio) DN:	955520 (0.67)	1495 (0.33)	10935 (0.99)
Total (Ratio) DO:	378256 (0.27)	2696 (0.59)	58 (0.01)
Total (Ratio) PR:	79172 (0.06)	334 (0.07)	24 (0)
Total (Ratio) NR:	13395 (0.01)	64 (0.01)	2 (0)
VALIDATIONS			
Total Validations:	608982	3848	5256
Total (Ratio) Agree:	200174 (0.33)	1186 (0.31)	8 (0)
Ave Agree Time:	00:00:09	00:00:08	00:00:18
Total (Ratio) Disagree:	408808 (0.67)	2662 (0.69)	5248 (1)
Ave Disgree Time:	00:00:08	00:00:07	00:00:02
OTHER			
Total Skips:	51616	142	26
Skip per annotation:	0.04	0.03	0
Total Comments:	26593	229	0
Comment per annotation:	0.02	0.05	0

Figure 4.17: Screenshot of the player profiling screen, showing the game totals and averages (left), a good player profile (centre) and a bad player profile (right) taken from real game profiles. The bad player in this case was identified by the speed of annotations and the only responses were DN in Annotation Mode and Disagree in Validation Mode. The player later confessed to using automated form completion software.

Table 4.7: Results of system error filters over the baseline for G1 on PD and PDFB.

G1	PD					PDFB				
	n	Pr	Re	Ac	F	n	Pr	Re	Ac	F
Baseline	15,191	0.77	0.97	0.80	0.86	21,952	0.66	0.92	0.70	0.77
PR()	15,102	0.78	0.97	0.81	0.86	21,751	0.67	0.92	0.71	0.78
RT_{zero}	15,183	0.77	0.97	0.80	0.86	21,952	0.66	0.92	0.70	0.77
Outlier	13,727	0.83	0.97	0.85	0.90	16,556	0.78	0.92	0.82	0.86
F_{error}	13,693	0.83	0.97	0.85	0.90	16,364	0.80	0.96	0.82	0.87
	-9.9%	**+.06**	**-**	**+.05**	**+.04**	**-25.5%**	**+.14**	**+.04**	**+.12**	**+.10**

Table 4.8: Results of system error filters over the baseline for W1 on PD and PDFB.

W1	PD					PDFB				
	n	Pr	Re	Ac	F	n	Pr	Re	Ac	F
Baseline	22,984	0.70	0.97	0.71	0.81	44,032	0.63	0.88	0.68	0.73
PR()	22,258	0.73	0.97	0.73	0.83	43,549	0.64	0.88	0.68	0.74
RT_{zero}	22,833	0.70	0.97	0.71	0.81	44,032	0.63	0.88	0.68	0.73
Outlier	13,826	0.76	0.98	0.77	0.85	25,279	0.65	0.92	0.71	0.76
F_{error}	13,688	0.76	0.98	0.77	0.86	24,813	0.67	0.92	0.73	0.77
	-40.4%	**+.06**	**+.01**	**+.06**	**+.05**	**-43.6%**	**+.04**	**+.04**	**+.05**	**+.04**

- System errors (F_{error})
- Player workload ($F_{workload}$)
- Player rating (F_{rating})
- Decision response time (F_{time})

Filtering out system errors Three types of error were identified in the data that were considered system errors because the data or data source were not possible or what would be expected. This filter was applied to the data first because the decisions that were removed were not likely to have been created by a human working in an environment capable of producing a good decision.

1. Recording a **PR()** interpretation should be impossible to enter as a game interpretation and represents a bug in the game.
2. A time of 0 (zero) seconds for an annotation or validation decision (RT_{zero}) is not possible. Response time is looked at in more detail later; however, a player recording a response in less than 0.5 seconds is more likely to represent a system error (or spam response) than a human response.

Table 4.9: Results of workload filters over the baseline for G1 on PD and PDFB.

G1	PD					PDFB				
	n	Pr	Re	Ac	F	n	Pr	Re	Ac	F
Baseline	15,191	0.77	0.97	0.80	0.86	21,952	0.66	0.92	0.70	0.77
Zipf	15,183	0.77	0.97	0.80	0.86	21,951	0.66	0.92	0.70	0.77
Top 20%	14,913	0.78	0.97	0.81	0.86	21,939	0.66	0.92	0.70	0.77
Top 10%	14,712	0.78	0.97	0.81	0.87	21,880	0.66	0.92	0.70	0.77
Top 1%	13,426	0.78	0.98	0.81	0.87	19,618	0.66	0.92	0.70	0.77
F_{error+}	13,266	0.84	0.98	0.86	0.90	16,292	0.80	0.96	0.83	0.87
$F_{workload\_10\%}$	**-12.7%**	**+.07**	**+.01**	**+.06**	**+.04**	**-25.8%**	**+.14**	**+.04**	**+.13**	**+.10**

Table 4.10: Results of workload filters over the baseline for W1 on PD and PDFB.

W1	PD					PDFB				
	n	Pr	Re	Ac	F	n	Pr	Re	Ac	F
Baseline	22,984	0.70	0.97	0.71	0.81	44,032	0.63	0.88	0.68	0.73
Zipf	22,973	0.70	0.97	0.71	0.81	44,032	0.63	0.88	0.68	0.73
Top 20%	22,512	0.70	0.97	0.71	0.81	44,021	0.63	0.88	0.68	0.73
Top 10%	22,219	0.71	0.97	0.71	0.82	43,893	0.63	0.88	0.68	0.73
Top 1%	19,188	0.71	0.97	0.71	0.82	40,182	0.64	0.87	0.68	0.74
F_{error+}	13,015	0.77	0.98	0.78	0.86	24,684	0.67	0.92	0.73	0.78
$F_{workload\_10\%}$	**-43.4%**	**+.07**	**+.01**	**+.07**	**+.05**	**-44.0%**	**+.04**	**+.04**	**+.05**	**+.05**

3. A method of profiling players was developed for the game to detect unusual or **outlier** behaviour. The profiling compares a player's annotations, validations, skips, comments and response times against the average for the entire game (see Figure 4.17). Based on the profiles of confessed spammers *blbuc (946)* and *gully (1000)* unusual player behaviour was identified: selecting DN responses for almost 100% of markables as this was the most efficient way to spam the game. Another unusual profile was few DO responses compared to DN, such as *Johnnickel (779)* or *askrukt (5970)* which might indicate a technological issue with their system configuration rather than cheating or perhaps not understanding the game rules. Whether a problem with the system or an attempt to cheat the game, users with a proportion of DN responses greater than 90% or a proportion of DO responses below 10% were excluded with this filter.

Removing system-error decisions improves the quality, but with a considerable loss to the total annotations and validations, especially in W1. However, losing this data should not be a concern as they are either invalid or from a source likely to be a cheat, spammer or from a software combination that is not compatible with the game. There appears to be an equal amount of discarded work in W1 between interfaces; however, there is more discarded work generated by PDFB in G1 (which was mainly the work

Table 4.11: Results of rating filters over the baseline for G1 on PD and PDFB.

G1	PD					PDFB				
	n	Pr	Re	Ac	F	n	Pr	Re	Ac	F
Baseline	15,191	0.77	0.97	0.80	0.86	21,952	0.66	0.92	0.70	0.77
Rating>60	14,765	0.78	0.97	0.81	0.87	21,951	0.66	0.92	0.70	0.77
Rating>70	4,475	0.79	0.96	0.82	0.87	21,686	0.66	0.92	0.71	0.77
Rating>80	3,346	0.84	0.96	0.86	0.90	21,291	0.67	0.92	0.71	0.77
Rating>90	2,710	0.88	0.96	0.89	0.92	11,043	0.82	0.97	0.85	0.89
F_{error+}	13,267	0.85	0.97	0.86	0.91	16,363	0.80	0.96	0.82	0.87
F_{rating_60}	**-12.7%**	**+.08**	**-**	**+.06**	**+.05**	**-25.5%**	**+.14**	**+.04**	**+.12**	**+.10**

Table 4.12: Results of rating filters over the baseline for W1 on PD and PDFB.

W1	PD					PDFB				
	n	Pr	Re	Ac	F	n	Pr	Re	Ac	F
Baseline	22,984	0.70	0.97	0.71	0.81	44,032	0.63	0.88	0.68	0.73
Rating>60	22,161	0.71	0.96	0.72	0.82	44,002	0.63	0.88	0.68	0.73
Rating>70	14,537	0.74	0.97	0.74	0.84	43,183	0.63	0.88	0.68	0.74
Rating>80	7,352	0.81	0.98	0.81	0.88	41,973	0.64	0.88	0.68	0.74
Rating>90	3,776	0.84	0.98	0.85	0.91	9,805	0.75	0.95	0.79	0.83
F_{error+}	11,046	0.79	0.98	0.80	0.87	24,783	0.67	0.92	0.73	0.78
F_{rating_60}	**-52.0%**	**+.09**	**+.01**	**+.09**	**+.06**	**-43.8%**	**+.04**	**+.04**	**+.05**	**+.05**

of one player). Removing this work produces large improvements in the overall quality of the remaining decisions (see Table 4.7 and Table 4.8).

Filtering by workload Players who do more work (make an annotation or validation decision) and progress to higher levels in the game are more likely to understand the task better and, ultimately, provide higher-quality decisions (see Section 4.7.1). For this reason a workload filter was tested by only selecting the decisions of a certain proportion of the hardest-working players. These proportions were converted into a workload amount:

- Players with less than 20 work, based on the findings of **Zipfian** curve fitting[1];
- The **top 20%** of players, represented by players who did more than 120 work and have done 98.3% of the total work;
- The **top 10%** of players, represented by players who did more than 315 work and have done 96.6 % of the total work[2];

[1]This filtering level was included to test the assumption in Section 4.7.1 that low-workload players are fundamentally different to other players; however, the small amount of excluded data means this filtering will have very little impact on overall quality.

[2]This filter level was added to have a middle ground that increased the amount of work excluded, but kept more players.

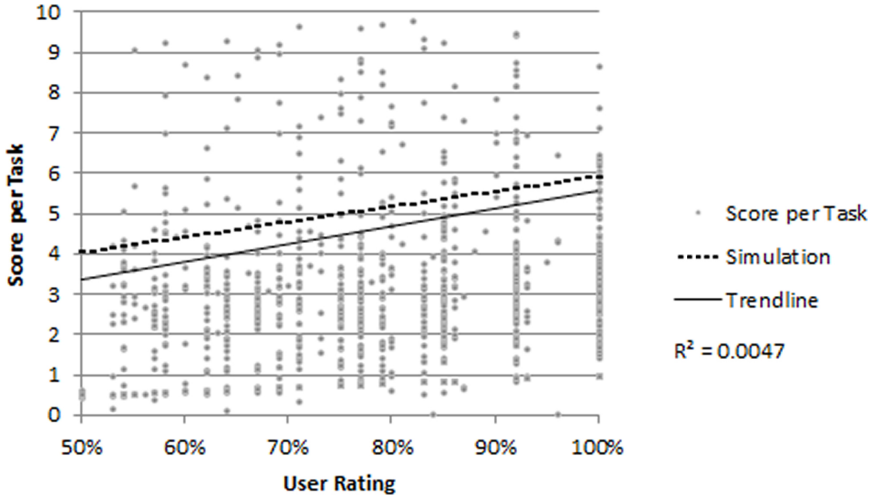


Figure 4.18: User-based correlation of score per task and rating, not showing outliers with more than ten points per task (P_{ub}=77.9 and I=2.3).

- The **top 1%** of players, represented by players who did more than 14,150 work and have have done 82.9% of the total work.

Filtering by workload does not appear to have much effect on the quality, partly because not many data were removed. By only using the data of the top 10% highest workload players ($F_{workload\_10\%}$) there is a small increase in quality without much data loss. Used in combination with the system error filters, this filter level shows a small improvement (see Table 4.9 and 4.10).

Filtering by player rating In order to test the assumption that higher-rated players should provide higher-quality decisions the AV Model (see Section 3.2.3) was simulated and tested against a subset of data from PDFB. The analysis uses coarse game data (total task-based score divided by the total number of tasks completed per user) and shows a weak correlation between score and user rating (n(1,329) R^2=0.005, Pearsons), with a similar slope gradient to the model when simulated using P_{ub} (the average rating of a player) and I (the average number of interpretations per markable) calculated from the dataset (see Figure 4.18). However, these data are incomplete as players may not have collected all the points for their work.[1] At this coarse level it appears that users

[1]Players should only be able to score a maximum of nine points per task (full disagreement in Validation Mode and then making a correction in Annotation Mode); however, a feature of the game

are rewarded in a way that the model would predict, i.e. higher-rated users provide better quality answers thus score more per task.[1]

Filtering based on players' ratings can have an effect on the quality, but at a cost to the quantity. In this context (post-collection ad-hoc data improvement) such large losses are not acceptable; however, in other contexts, when data quality is a priority, more extreme filters could be justified. Used in combination with the system error filters, increasing the rating threshold to 60% (F_{rating_60}) shows a small improvement with minimal additional data lost (see Table 4.11 and Table 4.12).

Filtering by response time One of the differences between *Phrase Detectives* and other games-with-a-purpose is that it uses pre-processing to offer the players a restricted choice of options. In Annotation Mode the text has embedded code that shows all selectable markables and in Validation Mode the player is offered a binary choice of agreeing or disagreeing with an interpretation. This makes the interface more game-like and allows reaction time to be investigated as a method of filtering in a more straightforward way as all responses are clicks rather than keyboard typing.

As motivation for why filtering by response time might improve the data quality an initial investigation was conducted to assess the types of responses each interface obtained. By using different types of data it was possible to identify three key stages of cognitive processing in the players to judge what a normal response might be (see Section 2.3.2).

The data analysed were from the first two years of data collection from each interface and does not include data from markables that are flagged as ignored. Responses of 0 seconds were not included because they were more likely to indicate a problem with the system rather than a sub 0.5 second response. Responses over 512 seconds (8:32 minutes)[2] were also not included and outliers do not represent more than 0.5% of the total responses.

A random sample of 50,000 responses per response type (annotation, agreeing validation, and disagreeing validation) shows that users respond differently between the

is that a player can skip or cancel the tasks they have been given. Any rewards from cancelled tasks are kept by the player, but not included in this calculation which explains the outliers.

[1]For further details of this filtering see Chamberlain [2014a].

[2]The upper time limit is set at 512 seconds because the data are part of a larger investigation that used RT grouped by a power function and it is assumed no task would take longer than this.

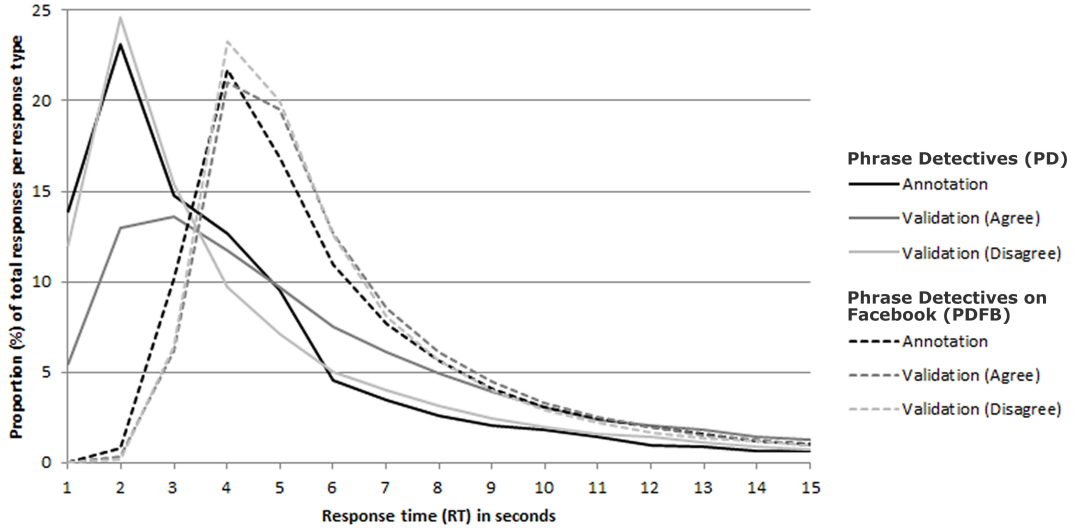


Figure 4.19: Proportional frequency of RT in the two modes of the two interfaces of *Phrase Detectives*.

two interfaces (n(150,000), $p<0.05$, unpaired t-test, see Table 4.13). The data were also plotted as a proportional frequency of RT, with a focus on the first 15 seconds (see Figure 4.19).

This may indicate a higher level of cheating and spam in PD; however, PDFB may be slower because it had to load the Facebook wrapper in addition to the interface. This is supported by RT_{min} for PDFB being 2.0s in Annotation and Validation (Disagree) Modes. The two interfaces differ in the proportion of responses two seconds or less (almost a third of all responses in PD, but a negligible amount in PDFB).

The RT for validations was slower than for annotations in the PD interface. This is counter-intuitive as Annotation Mode has more options for the user to choose from and requires a more complex motor response. One of the assumptions in the original game design was that a Validation Mode would be faster than an Annotation Mode and it would make data collection more efficient.

The analysis of cognitive stages of processing[1] supports the theory that filtering on reaction time would have a positive effect on quality. However, filtering on the minimum and maximum response times (RT_{min} and RT_{max}) does not improve the overall quality, in fact the former reduces quality in some cases. This is an indication that whilst RT

[1] A detailed analysis of player reaction times in *Phrase Detectives* is published elsewhere [Chamberlain and O'Reilly, 2014].

Table 4.13: Minimum, median and mean RT from a random sample of 50,000 responses of each response type from PD and PDFB.

	PD	**PDFB**
Annotation RT_{min}	1.0s	2.0s
Annotation RT_{med}	3.0s	6.0s
Annotation RT_{max}	7.2s	10.2s
Validation (Agr) RT_{min}	1.0s	1.0s
Validation (Agr) RT_{med}	5.0s	6.0s
Validation (Agr) RT_{max}	10.0s	10.5s
Validation (Dis) RT_{min}	1.0s	2.0s
Validation (Dis) RT_{med}	3.0s	6.0s
Validation (Dis) RT_{max}	8.4s	9.9s

could be used as an indicator of poor performance, players should be expected to take a range of times to complete an annotation task (see Tables 4.14 and 4.15). Based on these results response time was not used as a filtering method.

Combining filters to remove poor decisions The application of filters to the annotation and validation decisions increases quality (accuracy) between 5-13% in all the corpora; however, there is a large cost of annotations and validations that are discarded (between 16-55%) – see Table 4.16 and 4.17. The system error filters are likely to only be removing decisions that are spurious or malicious. The player workload and rating filters do improve the quality based on intuition and experimental observation, but may also be a case of overfitting the filter model to the data.

To assess whether overfitting was occurring the filtering model was applied to the GN, G2 and W2 corpora that, whilst smaller, could act as a test platform. The results show large improvements in quality over the baseline in all three corpora (see Tables 4.18, 4.19 and 4.20). The F_{error} filter does not appear to have as large an effect on these datasets, perhaps because they were annotated before players with system errors or spammers became involved in the game. The improvements indicate that the filters are not overfitted (at least within the *Phrase Detectives* system).

Table 4.14: Results of response time filters over the baseline for G1 on PD and PDFB.

G1	PD					PDFB				
	n	Pr	Re	Ac	F	n	Pr	Re	Ac	F
Baseline	15,191	0.77	0.97	0.80	0.86	21,952	0.66	0.92	0.70	0.77
RT_{min}>1s	14,863	0.77	0.97	0.80	0.86	21,951	0.66	0.92	0.70	0.77
RT_{min}>2s	13,522	0.77	0.97	0.80	0.86	21,877	0.66	0.92	0.70	0.77
RT_{min}>3s	11,571	0.78	0.97	0.80	0.86	20,459	0.66	0.93	0.70	0.77
RT_{min}>4s	9,464	0.78	0.97	0.81	0.86	15,458	0.65	0.92	0.70	0.76
RT_{max}<=180s	15,152	0.77	0.97	0.80	0.86	21,855	0.66	0.92	0.70	0.77
RT_{max}<=120s	15,119	0.77	0.97	0.80	0.86	21,816	0.66	0.92	0.70	0.77
RT_{max}<=60s	14,981	0.77	0.97	0.80	0.86	21,691	0.66	0.92	0.70	0.77
RT_{max}<=30s	14,640	0.78	0.97	0.81	0.86	21,306	0.67	0.93	0.71	0.78

Table 4.15: Results of response time filters over the baseline for W1 on PD and PDFB.

W1	PD					PDFB				
	n	Pr	Re	Ac	F	n	Pr	Re	Ac	F
Baseline	22,984	0.70	0.97	0.71	0.81	44,032	0.63	0.88	0.68	0.73
RT_{min}>1s	20,381	0.70	0.97	0.70	0.81	44,031	0.63	0.88	0.68	0.73
RT_{min}>2s	14,787	0.68	0.96	0.69	0.79	43,869	0.63	0.87	0.68	0.73
RT_{min}>3s	10,921	0.66	0.96	0.68	0.78	40,775	0.62	0.87	0.67	0.72
RT_{min}>4s	8,201	0.65	0.96	0.67	0.78	30,599	0.61	0.87	0.66	0.71
RT_{max}<=180s	22,859	0.70	0.97	0.71	0.81	43,851	0.63	0.88	0.68	0.73
RT_{max}<=120s	22,807	0.70	0.97	0.71	0.81	43,760	0.63	0.88	0.68	0.73
RT_{max}<=60s	22,593	0.70	0.97	0.71	0.81	43,459	0.63	0.88	0.68	0.73
RT_{max}<=30s	22,119	0.70	0.97	0.71	0.81	42,503	0.63	0.88	0.68	0.73

4.8.2 The influence of an expert in the crowd

Based on these results the players of the PD interface produce better-quality decisions, before and after filtering the data for poor-quality decisions. This is counter to what our initial hypothesis was (i.e. that social networks would encourage better-quality decisions from players) so a further investigation was conducted to see if the results were positively biased by collaborators in the project whose expertise and enthusiasm for playing the game mask the differences between the average game player.

All decisions from collaborators on the *Phrase Detectives* project[1] were removed from the data, but they did not represent enough data to make an impact (0.9% removed from G1 and 7.0% removed from W1). In the case of PD W1 6.6% of data were removed, leading to a small decrease in precision (-0.01) and accuracy (-0.01), but there are not enough data to assess the significance of this, although it is in line with what one would intuitively believe, i.e. that the experts in the crowd are improving the overall quality. Other than not providing enough data, the other reason that experts do not make an impact on quality is that the majority of tasks are not particularly

[1]Project collaborators were Jon Chamberlain (2), Massimo Poesio (18), Udo Kruschwitz (27) and Livio Robaldo (163).

Table 4.16: Improvements to decision quality over the baseline using filter combinations for G1 on PD and PDFB.

G1	PD					PDFB				
	n	Pr	Re	Ac	F	n	Pr	Re	Ac	F
Baseline	15,191	0.77	0.97	0.80	0.86	21,952	0.66	0.92	0.70	0.77
F_{error}	13,693	0.83	0.97	0.85	0.90	16,364	0.80	0.96	0.82	0.87
	-9.9%	**+.06**	**-**	**+.05**	**+.04**	**-25.5%**	**+.14**	**+.04**	**+.12**	**+.10**
F_{error}+	13,266	0.84	0.98	0.86	0.90	16,292	0.80	0.96	0.83	0.87
$F_{workload\_10\%}$	**-12.7%**	**+.07**	**+.01**	**+.06**	**+.04**	**-25.8%**	**+.14**	**+.04**	**+.13**	**+.10**
F_{error}+	13,267	0.85	0.97	0.86	0.91	16,363	0.80	0.96	0.82	0.87
F_{rating_60}	**-12.7%**	**+.08**	**-**	**+.06**	**+.05**	**-25.5%**	**+.14**	**+.04**	**+.12**	**+.10**
F_{error}+	12,840	0.86	0.97	0.87	0.91	16,292	0.80	0.96	0.83	0.87
F_{rating_60}+	**-15.5%**	**+.09**	**-**	**+.07**	**+.05**	**-25.8%**	**+.14**	**+.04**	**+.13**	**+.10**
$F_{workload\_10\%}$										

Table 4.17: Improvements to decision quality over the baseline using filter combinations for W1 on PD and PDFB.

W1	PD					PDFB				
	n	Pr	Re	Ac	F	n	Pr	Re	Ac	F
Baseline	22,984	0.70	0.97	0.71	0.81	44,032	0.63	0.88	0.68	0.73
F_{error}	13,688	0.76	0.98	0.77	0.86	24,813	0.67	0.92	0.73	0.77
	-40.4%	**+.06**	**+.01**	**+.06**	**+.05**	**-43.6%**	**+.04**	**+.04**	**+.05**	**+.04**
F_{error}+	13,015	0.77	0.98	0.78	0.86	24,684	0.67	0.92	0.73	0.78
$F_{workload\_10\%}$	**-43.4%**	**+.07**	**+.01**	**+.07**	**+.05**	**-44.0%**	**+.04**	**+.04**	**+.05**	**+.05**
F_{error}+	11,046	0.79	0.98	0.80	0.87	24,783	0.67	0.92	0.73	0.78
F_{rating_60}	**-52.0%**	**+.09**	**+.01**	**+.09**	**+.06**	**-43.8%**	**+.04**	**+.04**	**+.05**	**+.05**
F_{error}+	10,373	0.80	0.98	0.81	0.88	24,654	0.67	0.92	0.73	0.78
F_{rating_60}+	**-54.9%**	**+.10**	**+.01**	**+.10**	**+.07**	**-44.0%**	**+.04**	**+.04**	**+.05**	**+.05**
$F_{workload\_10\%}$										

hard or ambiguous therefore the skill of the expert is largely not required. Issues of task difficulty are explored in Section 5.4.

Although there were not enough data removed in this case to have an influence, the impact of a single player should not be underestimated given the observed Zipfian distribution of workload (see Section 4.7.1).

4.9 Credibility of player decisions

If the player rating is a good assessment of a player's ability to complete tasks, then the combination of ratings from the players that annotated a markable could be used as an assessment of credibility, i.e. how much we believe the interpretation is correct. During training all players are tested against a small set of tasks with known answers, with the proportion of answers correct creating the user's rating (see Section 4.5.2).

As a broad assessment of whether player rating is useful to determine correct answers, every decision made by players in the gold standard was assessed using the

Table 4.18: Improvements to decision quality using filter combinations for GNOME.

GNOME	**n**	**Pr**	**Re**	**Ac**	**F**
Baseline	5,597	0.71	0.97	0.75	0.82
F_{error}	4,607	0.75	0.98	0.80	0.85
	-17.7%	**+.04**	**+.01**	**+.05**	**+.03**
$F_{error}+$	3,579	0.82	0.99	0.85	0.89
$F_{rating\_60}+$	**-36.1%**	**+.11**	**+.02**	**+.10**	**+.07**
$F_{workload\_10\%}$					

Table 4.19: Improvements to decision quality using filter combinations for G2.

G2	**n**	**Pr**	**Re**	**Ac**	**F**
Baseline	2,035	0.67	0.94	0.74	0.78
F_{error}	1,571	0.76	0.97	0.81	0.85
	-22.8%	**+.09**	**+.03**	**+.07**	**+.07**
$F_{error}+$	1,377	0.81	0.98	0.86	0.89
$F_{rating\_60}+$	**-32.3%**	**+.14**	**+.04**	**+.12**	**+.11**
$F_{workload\_10\%}$					

Table 4.20: Improvements to decision quality using filter combinations for W2.

W2	**n**	**Pr**	**Re**	**Ac**	**F**
Baseline	5,877	0.57	0.95	0.65	0.71
F_{error}	5,138	0.59	0.95	0.67	0.73
	-12.6%	**+.02**	**-**	**+.02**	**+.02**
$F_{error}+$	3,260	0.69	0.96	0.75	0.81
$F_{rating\_60}+$	**-44.5%**	**+.12**	**+.01**	**+.10**	**+.10**
$F_{workload\_10\%}$					

Table 4.21: A table showing the difference in mean player rating for correct (true) and incorrect (false) decisions in both interfaces of *Phrase Detectives*. All differences between true and false answers and between interfaces were significant ($p<0.01$ unpaired t-test).

	TRUE	FALSE
G1 PD	72.4 sd(13.3) n(12,177)	69.5 sd(10.7) n(3,018)
G1 PDFB	91.4 sd(7.8) n(15,419)	86.5 sd(7.4) n(6,537)
W1 PD	75.7 sd(12.1) n(16,270)	71.8 sd(10.8) n(6,718)
W1 PDFB	87.0 sd(7.1) n(29,781)	84.6 sd(6.9) n(14,255)

precision and recall method (outlined in Section 4.8.1). Correct answers have a higher player rating than incorrect answers (see Table 4.21), with the implication being that player rating can be used as a measure of credibility. Additionally PDFB players had higher mean ratings than PD, perhaps explained best because PDFB players were tested frequently and therefore their rating more accurately reflects the improvement in ability since the first training session. PD only tests the rating once.

The credibility of a player's answer is a combination of a number of factors, including their internal knowledge, their skill at completing the task as intended by the designers of the system and the limit of their concentration. With anaphoric language annotation there is little internal knowledge required, in fact the annotation scheme specifically states that knowledge not in the context of the text displayed is not to be used, so this element should be consistent between players and tasks. The player's skill over time should increase and this would be reflected in improved ratings.

4.10 Summary

This chapter investigated whether a problem-solving system deployed on a social network can gather more answers of a higher quality than a standalone system. This question was investigated using the *Phrase Detectives* game-with-a-purpose that uses the players to validate the annotations, as well as enter the annotations themselves, with one system a standalone game (PD) and another deployed on the social network Facebook (PDFB).

Since the first release of the game on 1 December 2008 to 30 November 2014 (six years) 524 documents have been fully annotated, for a total completed corpus of 302,224

words and 95,415 markables. 38,594 players have registered, 2,746 of which went beyond the initial training phase. These players did more than 5,000 hours of work, i.e. 2.5 person-years and produced over three million annotations and validations.

The first investigation looked into how the players interacted on the different systems to see whether there was a difference in the quantity of work:

- PDFB had a higher conversion rate of users to trained players;
- Player workload followed a Zipfian distribution;
- PD's player recruitment was more successful than PDFB; however, the conversion of these users to trained players suggests this was due to an increase in spam registrations;
- PD had more active players per month than PDFB; however, PDFB active players did more work;
- There was a strong correlation between offering financial rewards and generating more work from the players;
- The players preferred the PDFB interface.

The quality of the players' decisions were then compared between PD and PDFB to investigate whether deploying a system on a social network will result in higher quality decisions:

- Both systems collected high-quality decisions; however, PD's decisions were of a higher quality than PDFB;
- There was a difference in the player response times between interfaces with PD being faster than PDFB;
- The player response time for validations was slower than for annotations in the PD interface;
- Filtering by response times did not improve the overall quality;
- The application of filters to the annotation and validation decisions increased quality (accuracy) between 5-13% in all the corpora; however, there was a large amount of work discarded (between 16-55%);

- Correct answers had a higher player rating than incorrect answers, with the implication being that player rating could be used a measure of credibility. Additionally, PDFB player decisions had a higher rating than PD player decisions.

The answer to the research question is multi-faceted but, in summary, players preferred using the game deployed on a social network and do more work, but in this case it does not translate to higher quality. Both versions of the game produce annotations and validation decisions close to what an expert would say, but this is at a high cost with considerable noise. The next section looks at how the process can be optimised.

5

AV Model: Optimising human effort with validation

It is a well-studied observation that a group of non-experts can perform as well, if not better, than a single expert at problem solving (see Section 2); however, can a more sophisticated model be used when the collected decisions are also validated by the crowd? This raises the question of whether gathering more opinions would be as valuable as validating existing opinions. Additionally, the question of answer confidence is raised, in particular problems in which there may be more than one possible solution (or no best solution). These issues are investigated using the AV Model proposed in Section 3.2.3 and evaluated in the *Phrase Detectives* game detailed in Section 4.5.

This chapter outlines how a baseline level of quality is established by first comparing the agreement between decisions from two experts, then by comparing an expert with the best answer from a system determined using the AV Model scoring (see Section 3.2.3). The data from *Phrase Detectives* are manipulated to compare scenarios that answer the following questions:

- Is a validation stage more efficient and able to produce better-quality answers than simply adding more annotations?
- What is the optimal configuration to reduce noise and increase efficiency?
- How confident can we be that we have the best answer and does task difficulty have an impact on the implementation of the model?

Portions of this chapter previously appeared in Poesio et al. [2013].

Table 5.1: Summary of gold standard datasets that are used in the data analysis, including total documents (D), total words (W) and total, unedited markables (M).

ID	Source	Description	*D*	*W*	*M*
GN	GNOME	Existing GS; PD only	5	874	274
W2	Wikipedia	2 expert GS; PD only	5	495	185
G2	Gutenberg	2 expert GS; PD only	1	180	69
W1	Wikipedia (combined)	1 expert GS	30	12,106	3,953
G1	Gutenberg	1 expert GS	4	6,231	1,971

5.1 Determining the quality of the best answer

In order to investigate the quality of annotations subsets of the corpora were used (see Table 4.1), each containing completed documents (see Table 5.1). The first subset was the annotated documents of the GNOME corpus (GN) which already had a documented gold standard [Poesio, 2004a]. The next subsets were the collection of documents from the Wikipedia (W2) and Gutenberg corpora (G2) that have a gold standard created by two experts.[1] These subsets were created on the PD interface. The final subsets were the Wikipedia (W1) and Gutenberg (G1) corpora with a gold standard created by one expert. These documents were selected at random from completed documents that had at least 50% of work done in the PDFB interface. See Appendix D for more details about how the gold standard was created for these corpora.

The player annotations can be examined at three levels of granularity: class; entity and specific. At the **class** level, a markable can be assigned one of four broad definitions (as previously defined in Section 4.4):

- DN (discourse-new): this markable refers to a newly introduced entity;
- DO (discourse-old): this markable refers to an already mentioned entity in the text;
- NR (non-referring): this markable does not refer to anything (e.g. pleonastic *it*);
- PR (property attribute): this markable represents a property of a previously mentioned entity (e.g. *a teacher* in 'He is a teacher').

[1] W2 was used for the initial investigation of quality [Chamberlain, Kruschwitz, and Poesio, 2009].

Table 5.2: Inter-expert agreement between e2 and e18 (DN = discourse-new, DO = discourse-old, NR = non-referring, PR = property attribute).

	GN n(59)	**W2** n(154)	**G2** n(57)
DN	-	99.0%	85.7%
DO	96.6%	87.8%	97.2%
DO (specific)	93.2%	84.8%	91.6%
NR	-	100%	-
PR	-	72.7%	-
PR (specific)	-	72.7%	-
Overall (specific)	93.2% (κ=0.93)	94.1% (κ=0.88)	89.4% (κ=0.88)

At the **entity** level the two classes DO and PR allow for a referring entity to be selected, for example, *he* referring to the entity *Dave* in 'Dave was the best he could be.' At the **specific** level the closest mention of the entity in the text in terms of character distance from the markable is considered correct which allows for linear anaphoric chaining to occur. A correct example would be *she* referring to the markable *her* in 'Kate wondered if her suit was the best she had.' which are both mentions of the entity *Kate*.

The game's design and player instructions allow for class and specific annotations to be collected. Unless otherwise stated, the specific level of annotation granularity is analysed in the results.

5.1.1 Agreement between expert annotators

One way to tell whether the game was successful at obtaining good quality anaphoric annotations was to check how the aggregated annotations produced by the game compare to those produced by an expert annotator and it is also useful to know what is the agreement between two experts annotating those texts.

Five completed documents from the Wikipedia corpus containing 154 active markables (W2) and one document from the Gutenberg corpus containing 57 active markables (G2) were selected. Each document was manually annotated by two experts operating independently[1] (see Appendix D for details about how the gold standard

[1]The two experts were Jon Chamberlain (who developed the game and wrote the instructions) and

was created).

The five documents from the GNOME (GN) corpus were annotated by e2 and compared to the consolidated annotations of the GNOME corpus (of which e18 was the main annotator).[1] DN and PR annotations were not recorded and there were no instances of NR markables. The GNOME annotations were recorded in *Phrase Detectives* under the expert ID e39181. In total there were 59 markables that e2 and GNOME produced an annotation for (see Table 5.2).

Overall, agreement between experts in the three corpora was very high although not complete: 93.2% (GN), 94.1% (W2) and 89.4% (G2), for a chance-adjusted κ value [Artstein and Poesio, 2008] of $\kappa = .93$, $\kappa = .88$ and $\kappa = .88$ respectively, which is extremely good. This value can be seen as an upper boundary on what we might get out of the game.

There was no significant difference between the inter-expert agreement of the three corpora (GN n(59) 93.2%; W2 n(154) 94.1%; G2 n(57) 89.4%; p=0.810, p=0.238, p=0.465, z-test) which shows that the expert annotations created by e2 are what could be considered a gold standard when compared to an existing gold standard and another linguistic expert. Expert annotator e2 also created the gold standard for W1 and G1.

5.1.2 Baseline measures of agreement

Traditional methods of measuring annotation generally assume a singularity of correct answers, but measuring accuracy of a multi-dimensional annotation set is more complex. In this section, the best answer from the game was used as an accuracy measure and incorrect assignments were further investigated for ambiguity.

The performance of the game was measured by four variables: quality; cost; noise; and speed. These variables are of consideration when testing aggregation models to assess quality as well as to reduce the cost, noise and speed of getting a crowd answer.

Quality is measured as the level of agreement between an expert and the highest-scoring system answer.

Noise is defined as the number of wrong interpretations per markable.

Massimo Poesio (a linguistic expert in anaphoric coreference), called e2 and e18 respectively in the rest of this discussion.

[1]The GNOME annotation scheme only records DO annotations (as 'ident' variables) and plural DO was only recorded once (as an 'element-inv' variable) so ignored here.

Table 5.3: Baseline agreement between the two experts and the best answer from the game.

	GN		**W2**		**G2**	
	e2	e39181	e2	e18	e2	e18
Markables	264	61	176	160	63	58
Agreement	93.9%	85.2%	84.0%	81.8%	96.8%	93.1%
Kappa κ	0.86	0.85	0.63	0.59	0.96	0.92
$Noise_{mean}$	1.6		2.7		2.6	
	sd(2.0)		sd(3.4)		sd(2.1)	
$Cost_{mean}$	21.6		31.5		31.8	
	sd(15.0)		sd(22.9)		sd(16.3)	
$Speed_{mean}$	308.2		544.9		286.1	
$Speed_{median}$	155		276		189	
	sd(471.1)		sd(783.2)		sd(304.4)	

Cost is measured as the total number of annotations and validations (work) that are required to produce an answer set per markable.

Speed is defined as the time (in seconds) to create the game answer by summing all the response times of the annotations and validations per markable.

The annotations and validations of each markable from each corpus were analysed and either aggregated to produce a best answer or were excluded because:

- the markable has been marked by an administrator to be ignored;
- the expert did not provide an answer (therefore an answer was not possible);
- the markable was skipped by enough players (the markable does not have eight annotations).

In the baseline AV Model all annotations and validations for each interpretation of a markable were combined:

$$A + V_a - V_d$$

A is the number of players initially choosing the interpretation in Annotation Mode, V_a is the number of players agreeing with that interpretation in Validation Mode, and

Table 5.4: Baseline agreement between the expert e2 and the best answer from the game in the G1 corpus.

G1 n(1,844)	$A + V_a - V_d$	A_8	$A_8 + V_a - V_d$	$A_8 + V_a$	$A_8 - V_d$
Agreement	86.6%	78.5%	86.0%	85.3%	85.2%
Kappa κ	0.85				
$Noise_{mean}$	1.4	1.2	1.1	1.1	1.1
	sd(1.3)	sd(1.0)	sd(1.0)	sd(1.0)	sd(1.0)
$Cost_{mean}$	20.3	8	14.8	10.9	11.9
	sd(10.1)	sd(0)	sd(4.7)	sd(2.2)	sd(3.2)
$Speed_{mean}$	231.2	96.2	172.2	130.6	137.8
$Speed_{median}$	152	64	116	86	92
	sd(448.9)	sd(259.5)	sd(357.3)	sd(300.3)	sd(322.2)

V_d is the number of players disagreeing with it in Validation Mode. This formula is used to score each interpretation of a markable, with the highest scoring interpretation called the 'best' or game interpretation.

The baseline agreement in the three corpora in which two experts provided a gold standard show very high agreement, comparable to pairwise inter-expert agreement (see Table 5.3). The best game answer more frequently agreed with e2, most likely because e2 wrote the instructions for the game players; however, this was only statistically significant (GN, p=0.02; W2, p=0.59; G2, p=0.35; z-test) in the GNOME corpus for which the annotation scheme was different (e39181 only annotated DO).

Both W1 and G1 have lower agreement (quality) than W2 and G2, significantly so in the Gutenberg corpus (G1-G2, z-test, p=0.02; W1-W2, z-test, p=0.12) which may be because the latter documents were worked on by more linguists and friends of the researchers, rather than the former documents which were worked on by a real crowd of unknown people, or perhaps an artefact of outliers in the crowd (see Section 4.8.2). This may also explain the difference in the performance of the two interfaces (see Section 4.8).

The baseline figures for the five gold standard corpora show high quality at near-expert annotator performance; however, the cost, noise and speed are high making this method too expensive via microworking, too noisy for extracting data in high-spam scenarios and too slow for short-term data collection projects.

Table 5.5: Baseline agreement between the expert e2 and the best answer from the game in the W1 corpus.

W1 n(3,729)	$A + V_a - V_d$	A_8	$A_8 + V_a - V_d$	$A_8 + V_a$	$A_8 - V_d$
Agreement	79.1%	74.2%	79.2%	77.6%	77.5%
Kappa κ	0.52				
$Noise_{mean}$	1.3	1.1	1.0	1.0	1.0
	sd(1.6)	sd(1.2)	sd(1.1)	sd(1.1)	sd(1.1)
$Cost_{mean}$	18.7	8	13.2	9.9	11.3
	sd(12.0)	sd(0)	sd(5.2)	sd(2.1)	sd(3.7)
$Speed_{mean}$	234.8	97.0	171.9	122.8	146.0
$Speed_{median}$	121	51	92	66	75
	sd(1,068.9)	sd(797.4)	sd(1,046.0)	sd(846.0)	sd(1,007.1)

Resolving tied results There were occasions when the game produced two interpretations of an equally high score. The W1 and G1 corpora were tested to see the difference in agreement should the first answer entered in the game be preferred or the most recent answer. In both cases, the agreement was slightly higher when preferring the most recent interpretation; however, this was not a significant difference (G1 n(1,844) oldest first 86.5%, newest 86.6%, z-test p=0.92; W1 n(3,729), oldest 78.7%, newest 79.1%, z-test, p=0.67). This may not be such an issue when using more complex aggregation techniques, as a draw is less likely.

5.1.3 How many annotators are required to match an expert?

With a majority voting scheme there is an assumption that the larger the crowd, the more chance there is of getting the best answer to be in agreement (in this case) with an expert, which is the approach of microworking. It is assumed that several annotators are superior to a single annotator, which is the approach of traditional, partly-validated expert annotation. The questions raised in Section 3.2.1 were how many annotators are enough and when to stop collecting annotations.

The expectation of declining returns from adding annotators past a certain point is tested by comparing the agreement in the W1 and G1 corpora by using increasing numbers of annotators (in date order, oldest first – see Figure 5.1). There is only a small increase in agreement in the W1 corpus between one and eight annotators (A_1

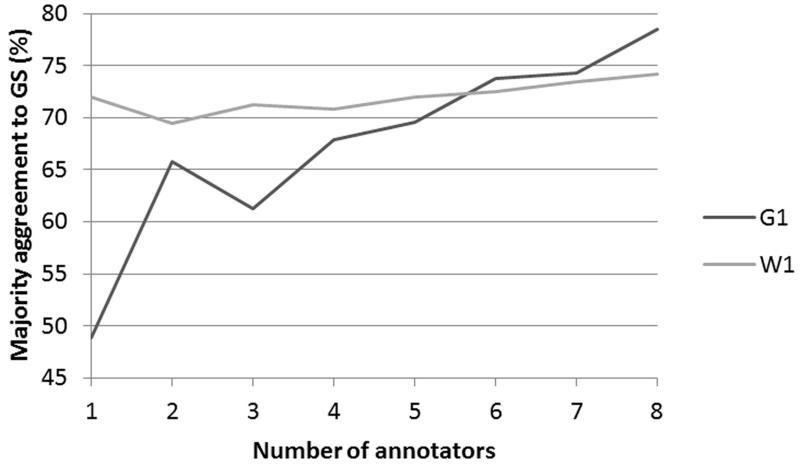


Figure 5.1: Chart showing the majority voting agreement to the gold standard for different numbers of annotators for G1 and W1.

to A_8), whereas the G1 corpus has a very large and incremental increase of agreement. The latter may be caused by poorer annotations in the G1 corpus and with more annotations the poor decisions become voted out to approach the true upper limit of agreement (see Table 5.4 and 5.5).

5.1.4 Improving annotation with validation

By adding the validation step to the eight annotations ($A_8 + V_a - V_d$), there is a significant increase in agreement in both corpora (G1 and W1, $p<0.01$, z-test, see Tables 5.4 and 5.5), whilst noise is not affected (as validation only votes up or down an interpretation).[1] The validation step will of course increase the cost and the time to complete the markable (see Table 5.4 and 5.5).

The results show that the validation stage can **increase the overall quality of a crowd system without introducing more noise**.

Does validation replace the need for filtering data? The filtering methods were applied to the baseline aggregation techniques and whilst it did increase the agreement in four of the five corpora (GN had no change) the change was not significant (G1 n(1,804) 86.6% pre-filtered, 88.9% post-filtered, p=0.03, z-test; W1 n(3,729) pre-filtered

[1]There is actually a slight difference in noise caused by some of the markables not being included in the calulations because they did not fully complete the validation stage.

Table 5.6: For all corpora the number of annotations required before the correct interpretation is introduced to the answer set would be 6.

	n	Mean	SD	Upper 2 SD
W1	3,534	1.99	2.38	6.75
G1	1,800	2.01	1.69	5.39

79.1% post-filtered 80.1%, p=0.285, z-test). This is an indication that the aggregation methods used in the AV Model are an effective, if not cost-efficient, way to remove spurious or malicious interpretations (see Table 5.9).

5.2 Optimising the AV Model

5.2.1 Do we need to disagree?

We have seen that using validation can significantly increase the quality of crowd aggregated answers. It is quite common on thread-based or QA websites to see validation or voting buttons, but some may only have an upvote or 'like' button, the most notable example being Facebook. Here we test whether the same increase in agreement could be achieved by only using agreement validation (V_a) decisions.

On both G1 and W1 corpora there is no significant difference in agreement between full validation and either using agreement ($A_8{+}V_a$) or disagreement ($A_8{-}V_d$) validations (G1 n(1,844) p=0.542 (Va) p=0.490 (Vd), z-test; W1 n(3,729) p=0.093 (Va) p=0.075 (Vd), z-test), see Tables 5.4 and 5.5. This means that a system that uses agreement validation or a like/upvote button such as Facebook can achieve the same level of quality for significantly less effort and time than using disagreement or full validation.

5.2.2 Completeness vs noise

In the *Phrase Detectives* game, the only way to add additional interpretations after the initial eight annotations was to disagree with an interpretation. This allows the markable to be annotated indefinitely until all users have entered their preferred interpretations.

However, in order to find a suitable stopping point for aggregation we determine how few annotations are required before most markables have been given the correct

Table 5.7: Optimised agreement between the expert e2 and the top answer from the game.

G1 n(1,844)	$A + V_a - V_d$	$A_6 + V_a$	$A_6 + V_a$ filtered
Agreement	86.6%	84.1%	88.9%
Kappa κ	0.85		
$Noise_{mean}$	1.4	1.0	0.6
	sd(1.3)	sd(0.9)	sd(0.9)
$Cost_{mean}$	20.3	8.7	7.1
	sd(10.1)	sd(2.1)	sd(2.2)
$Speed_{mean}$	231.2	108.3	78.6
$Speed_{median}$	152	67	53
	sd(448.9)	sd(293.8)	sd(157.8)

answer. Each gold standard markable (when the correct interpretation was within the answer set) was measured to see how many annotations were required before the gold standard interpretation was introduced. This was averaged across all the markables in each corpus. By calculating the distance from the mean of two standard deviations we can estimate that 97.5% of the markables will have the correct relation added to the answer set (although the data are non-parametric and constrained so this estimate may be inaccurate). According to these estimates, we require between five and seven annotations before the gold standard interpretation is added to 97.5% of markables in the W1 and G1 corpora (see Table 5.6).

From previous analysis in Section 5.1.4 we get similar levels of agreement for the full AV Model ($A + V_a - V_d$) and a restricted Model ($A_8 + V_a - V_d$) (G1, 86.6% vs 86.0% n(1,844), p=0.596, z-test; W1, 79.1% vs 79.2%, n(3,729), p=0.912 z-test). Knowing most of the interpretations should be captured within six annotations and therefore further annotations were likely to introduce more noise, an optimised model ($A_6 + V_a$) was tested and showed agreement was not significantly reduced (G1 n(1,844) full 86.6%, optimised 84.1%, p=0.03, z-test; W1 n(3,729) full 79.1% optimised 76.9, p=0.02, z-test), but the noise and cost were (see Table 5.7 and 5.8).

Using weighted aggregation A player's rating should be a good indicator of how good they are at solving the problem so their answers could be considered more credible.

Table 5.8: Optimised agreement between the expert e2 and the top answer from the game.

W1 n(3,729)	$A + V_a - V_d$	$A_6 + V_a$	$A_6 + V_a$ filtered
Agreement	79.1%	76.9%	80.1%
Kappa κ	0.52		
$Noise_{mean}$	1.3	0.8	0.7
	sd(1.6)	sd(1.0)	sd(1.0)
$Cost_{mean}$	18.7	7.4	5.9
	sd(12.0)	sd(1.9)	sd(2.9)
$Speed_{mean}$	234.8	93.9	61.1
$Speed_{median}$	121	47	33
	sd(1,068.9)	sd(807.0)	sd(230.5)

Using the player's rating for aggregated scoring instead of an integer count allows the judgements of better-performing players to have more impact than those with poorer performance, although it will be biased in favour of PDFB players because their ratings will increase over time. It also has the advantage of resolving tied results between multiple answers.

Using weighted scoring instead of integer scoring on the baseline AV Model ($A + V_a - V_d$) does not show significant improvement in agreement (G1, 86.5% from 86.6%, p=0.596 z-test; W1 80.1% from 79.1%, p=0.284 z-test). When compared on the optimised model ($A_6 + V_a$) there was also no significant improvement (G1 83.4% from 84.1%, p=0.562 z-test; W1 77.8% from 76.9%, p=0.352 z-test).

These results show that using a weighted aggregation method has no effect on agreement scoring and that the aggregation method itself is more powerful. However, the weighting could still be used as a way to assess the confidence in a game answer comparatively.

5.2.3 The optimised and filtered AV Model

The optimised model was also filtered (as in Section 4.8.1) and, unlike the full AV Model, was improved, with the agreement improved over the baseline, in addition to the reduced noise, cost and increased speed (see Tables 5.7 and 5.8). With simple adjustments to the system (represented by the filtering), along with the optimised model,

Table 5.9: Summary of agreement under different AV Model conditions, showing that the optimised and filtered AV Model performs as well as the full baseline AV Model.

	GN	**G2**	**W2**	**G1**	**W1**
Markables	275	63	176	1,884	3,729
Inter-expert	93.2%	89.4%	94.1%		
Baseline agreement ($A + V_a - V_d$)	93.9%	96.8%	84.0%	86.6%	79.1%
Baseline+filtered ($A + V_a - V_d$)	93.9%	98.4%	85.2%	88.5%	79.4%
Optimised ($A_6 + V_a$)	93.1%	96.8%	81.2%	84.1%	76.9%
Optimised+filtered ($A_6 + V_a$)	93.5%	98.4%	84.6%	88.5%	80.1%
Difference over baseline	**-0.4%**	**+1.6%**	**+0.6%**	**+2.3%**	**+1.0%**
p (z-test)	0.849	0.555	0.881	0.077	0.285

dramatic improvements to system performance can be achieved in all four key criteria. A summary of quality under different conditions for the five corpora is presented in Table 5.9.

5.3 Confidence in the best answer

The confidence in a game answer is a product of the credibility of the users that gave each annotation. For cases in which there is one correct interpretation, the more annotations from credible sources that choose the same answer, the more confident we can be this is the best answer. Ambiguous cases (when there is more than one possible interpretation) can be detected by having these interpretations supported by credible users. In the *Phrase Detectives* system, all markables were treated the same way with no dynamic stopping rules in place; however, we can investigate whether the confidence of correct answers are distinguishable from incorrect answers and whether it is possible to detect multiple correct answers using confidence scoring.

In both the W1 and G1 corpora the best interpretation from the game has a significantly higher confidence score than an incorrect interpretation ($p<0.01$, unpaired t-test – see Table 5.10).

There are very few cases of genuine ambiguity in the W1 and G1 corpora (marked by the expert as 'possible') so it is hard to draw a conclusion as to whether these interpretations could be extracted automatically. There were more examples of inter-

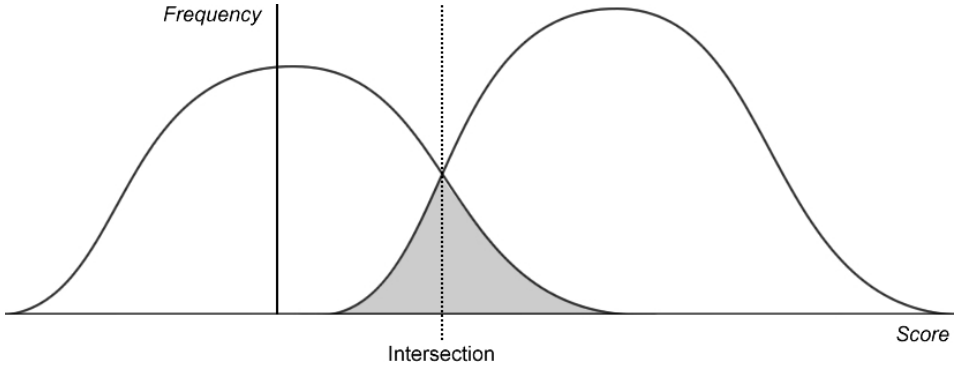


Figure 5.2: An example of a confidence graph showing a distribution of correct answer scores (right curve) and incorrect scores (left curve). The point where they intersect would be the ideal threshold for filtering interpretations; however, the grey area under the graph (the overlap) shows what proportion of the data would be uncertain in that condition.

pretations which refer to the correct entity but were not the closest mention (marked by the expert as 'same entity'); however, these have a lower score than wrong answers and would not be possible to automatically extract using this method. In this system interpretations that are the same entity but not the closest mention would in fact be considered incorrect as this was specifically stated in the instructions for the players and in the expert annotation scheme.

The difference between the best interpretation and an incorrect interpretation can be considered as two overlapping normal distributions (see Figure 5.2). The point where the distributions intersect could be used as a threshold to determine whether an interpretation is more likely to be correct or incorrect. The grey area of the graph highlights the overlap of the two distributions and represents the proportion of interpretations that could not be judged in this way. Whilst there is a distinct difference between the scores of correct and incorrect interpretations the degree of overlap shows that a considerable proportion of interpretations cannot be judged by their score (23.6% for G1 and 34.3% for W1 – see Table 5.10). When using the optimised AV Model ($A_6 + V_a$) the area of overlap is even larger (30.4% for G1 and 47.6% for W1). This indicates a potential disadvantage of using the optimised AV Model in that there would be less confidence in the best answer from the system.

Both AV Models were tested to see if weighted answers would increase the confidence of the best answer (i.e. would reduce the area under the curve); however, in all cases

Table 5.10: A table showing the difference in confidence of the correct answer against incorrect answers in *Phrase Detectives*.

	n	mn	sd	min	med	max	intersect	overlap
G1 $A + V_a - V_d$							5.6	23.6%
Gold standard	1,814	10.5	4.3	-3	10	32		
Incorrect	1,979	0.8	3.9	-3	0	20		
Possible	9	2.7	5.5	-3	2	12		
Same Entity	346	-0.2	3.7	-3	-1	15		
G1 $A_6 + V_a$							4.6	30.4%
Gold standard	1,760	6.4	1.7	1	6	9		
Incorrect	1,553	2.8	1.8	1	2	9		
W1 $A + V_a - V_d$							5.1	34.3%
Gold standard	3,537	8.6	3.8	-3	8	28		
Incorrect	4,027	2.0	4.8	-3	1	29		
Possible	28	1.0	3.9	-3	0	14		
Same Entity	395	-0.6	2.8	-3	-1	14		
W1 $A_6 + V_a$							4.4	47.6%
Gold standard	3,300	5.8	1.4	1	6	9		
Incorrect	2,538	3.4	2.1	1	3	9		

there was little or no improvement in confidence (G1 full AV -0.5%, optimised -0.3%; W1 full AV -9.8% optimised +1.4%). This would suggest that weighted aggregation techniques are not as powerful as validation techniques for identifying the best answer.

5.4 Task distribution and difficulty

In this analysis all markables have been treated in the same way; however, it is clear that some markables are easier to annotate than others, either because of the text itself or because of the type of relation it has with the other markables.

Contextual difficulty It could be assumed that the more complex the text, the more difficult the users would find the task of annotating the markables, and therefore the quality would be lower. However, agreement per document shows a weak positive correlation to readability (n(45) R=0.19 R^2=0.037; Pearson, weak positive correlation)

Table 5.11: Summary of the distribution of interpretations for active markables in the gold standards as created by e2.

	DN	DO	NR	PR	NM
GN n(275)	189 (68.7%)	65 (23.6%)	0	4 (1.4%)	17 (6.1%)
W2 n(176)	128 (72.7%)	33 (18.7%)	1 (0.5%)	13 (7.3%)	1 (0.5%)
G2 n(63)	27 (42.8%)	36 (57.1%)	0	0	0
W1 n(3,729)	2,502 (67.0%)	912 (24.4%)	23 (0.6%)	108 (2.8%)	184 (4.9%)
G1 n(1,884)	638 (33.8%)	1,160 (61.5%)	25 (1.3%)	21 (1.1%)	40 (2.1%)

implying readability has little impact on the user's ability to perform annotation tasks. The Wikipedia documents in the corpus were not complex and more extreme examples in other documents might show different results.

The Gutenberg corpus has a higher agreement than the Wikipedia corpus (G1 n(1,844) 86.6%, W1 n(3,729) 79.1%, $p<0.01$, z-test), but also a higher noise rate (Mann Witney U-test, $p<0.01$), a higher cost (Mann Witney U-test, $p<0.01$) and slower median speed (Mann Witney U-test, $p<0.01$) (see Table 5.4 and 5.5). This supports the previous findings in the descriptive analysis that the narrative texts of Gutenberg are easier to annotate (see Appendix F) but require more thought.

To investigate whether document length has an impact on difficulty the W1 corpus was split into two groups, one with long documents (WL1, greater than 700 words long) and one with short documents (WS1, less than 700 words long). There was no difference between the agreement in the Wikipedia long and short corpora (WL1 n(1,947) 79.9%; WS1 n(1,782) 78.1%; z-test, $p=0.18$) which confirms that document length also does not seem to impact on a user's ability to annotate the text.

Interpretation difficulty In order to explore whether some types of interpretation were harder to detect and annotate than others, the classes of interpretation were first examined to see how they were distributed through the corpora, then at the agreement of each class.

The distribution of annotation class is calculated as a proportion of interpretations of active markables as determined by an expert (e2). When there was no correct interpretation the markable would be classed as NM (Not Mentioned), see Table 5.11.

Table 5.12: Breakdown of agreement between each interpretation type (as determined by e2) on the Wikipedia and Gutenberg corpora, showing a difference in all classes of interpretation ($p<0.01$, z-test).

	G1	W1
Markables	1,844	3,729
DN	91.5% (584 of 638)	98.5% (2,466 of 2,502)
DO (specific)	88.0% (1,021 of 1,160)	49.8% (455 of 912)
NR	96.0% (24 of 25)	65.2% (15 of 23)
PR (specific)	19.0% (4 of 21)	12.9% (14 of 108)
Overall agreement	86.6%	79.1%

The documents in G1 have more coreferring DO markables (61.5%) than in the documents in W1 (24.4%), with the reverse being true for DN markables. NR and PR markables are rare in both corpora (W1 n(3,729); G1 n(1,884); χ^2=763.6, $p<0.01$). One explanation might be that as Wikipedia articles, which are explanatory in nature, become longer they introduce more entities to explain the topic of the document. The reverse could be true for Gutenberg documents, that are mainly narratives, that will introduce entities and continue to refer to them throughout the discourse.

A closer look at the breakdown of agreement between the best game answer and e2 shows a significant difference between the performance of players on the Gutenberg and Wikipedia corpora on different tasks (see Table 5.12). The Wikipedia corpus had more DN and less DO markables (as determined by e2) than Gutenberg (see Table 5.11). These results show that DN is an easier task and as W1 has more true DN markables it could be expected that the W1 corpus would be annotated to a higher quality. However, this is not the case due to the poor performance of interpretations of DO markables in the W1 corpus. This shows that task difficulty has a considerable impact on the quality that can be achieved by a crowd.

Viewing the document as a whole, factors such as document length and readability do not seem to impact agreement; however, users do find it harder to detect and annotate different types of interpretation. This should be a consideration when estimating the confidence of an answer set, for example, if the best answer has been determined to be discourse-new there would be higher confidence that the users made the decision correctly and this was the true interpretation compared to annotating a property which

are difficult decisions.

5.5 Summary

This chapter investigated the AV Model implemented in the *Phrase Detectives* game to answer the question of whether a validation step can provide higher quality results than just acquiring more annotations.

By comparing the work of annotators against annotators with an additional validation step, the validation stage was shown to increase the overall quality of a crowd system without introducing more noise.

The baseline figures for the five gold standard corpora showed high quality at near-expert annotator performance; however, the cost, noise and speed were also high making this method too expensive via microworking, too noisy for extracting data in high-spam scenarios and too slow for short-term data collection projects.

The next question was whether the AV Model could be optimised to maintain quality but reduce noise and increase efficiency. The investigation showed that using agreement validation (instead of full validation or disagreement validation) does not reduce quality but increases efficiency. This means that a system that uses agreement validation or a like/upvote button (as we shall see implemented in Chapter 6) can achieve the same level of quality for significantly less effort and time.

Additionally, an optimised model reduced the number of annotations that were required, again not significantly affecting quality but reducing noise and cost. This reinforces the idea that understanding how many opinions need to be gathered before stopping is key to making a crowd-based system efficient.

In both the W1 and G1 corpora the correct interpretation from the game has a significantly higher confidence score than an incorrect interpretation so we can have high confidence in answers that score more; however, this confidence is lower in the optimised model. There were very few cases of genuine ambiguity in the corpora and the player instructions ensured that 'same entity' interpretations were considered as wrong answers. This makes automatically extracting these cases from the data very difficult.

Factors such as document length and readability do not seem to impact quality. However, users did find it harder to detect and annotate different types of interpretation,

and the frequency of difficult tasks within different document topics will influence the overall quality obtainable from a system.

6

Groupsourcing: Inherent problem solving on social networks

The study of *Phrase Detectives* and the AV Model (Chapters 4 and 5) show that deploying problem-solving systems on social networks has many benefits, but it is not clear whether they offer an improvement in already high-performing systems. The final investigation explores the idea that problem solving is an inherent part of the way humans interact with each other on social networks and that it can be viewed in the same way as a crowdsourcing system.

One of the most important findings from Chapter 4 was that whilst users of a system deployed on a social network are more active and engaged with the system, the quality they produce is not as high as a stand-alone system. There were several explanations for this, mainly due to the practicality of operating the two systems simultaneously. Here we investigate the quality of problem solving compared to experts and another common method of crowdsourcing, namely microworking in which the users are paid.

Chapter 5 showed that using agreement validation is a cost-effective and efficient way to improve the quality of data extracted. The AV Model is tested in a social network setting to understand whether agreement validation displays a similar effect on quality.

Portions of this chapter previously appeared in Chamberlain [2014b,c].

6.1 Introduction

The *Phrase Detectives* game, much like other methods of crowdsourcing, requires the users to complete preset tasks using a well-defined interface. The tasks are structured and the user response is captured in a constrained way. Whilst this method makes life easier for those collecting the data, it creates an unnatural interaction between the human and the system that means the users must be motivated to participate.

One alternative is to allow users to generate solutions to their own needs naturally, and this can been seen on social networks such as Facebook, Twitter, Flickr and LinkedIn with the evolution of groups to solve problems. Social network crowdsourcing is distinguished by several features: data and tasks are created by the users; input is unconstrained and developed in series whilst simultaneously validated by the users themselves; users are inherently motivated, socially trained and work collaboratively; and the output is immediately accessible and beneficial to all, with users receiving recognition for their efforts (see Section 3.1).

This chapter investigates groups on the social network Facebook in which the users attempt to identify and classify images of marine life.

6.2 Definitions

Groupsourcing is defined as *completing a task using a group of intrinsically-motivated people of varying expertise connected through a social network* (see Section 3.4). A **group** in this context is a feature of a social network that allows a small subset of users to communicate through a shared message system.

Groups are initially set up in response to the needs of a few people and the community evolves as news from the group is proliferated around the network in feeds and user activity.

The group title, description and 'pinned' posts usually give clear indications as to whom the group is aimed at and for what purpose. This research focuses on three types of group motivation that were considered likely to contain problem solving (the examples are from the domain of marine biology):

1. **Task Request (TR)** - groups in which users are encouraged to post messages with a task, e.g. *ID Please (Marine Creature Identification)*

Figure 6.1: Detail of a typical message containing an image classification task posted on a social network (in this case Facebook).

2. **Media Gallery (MG)** - groups in which users are encouraged to share media (image and video) for its artistic merit, e.g. *Underwater Macro Photographers*

3. **Knowledge Sharing (KS)** - groups used for coordination of activities or for distributing knowledge, research and news, e.g. *British Marine Life Study Society*

Groups can also be categorised into those that are **specific** to a topic or subject (-S) and those that are non-specific or **generalist** (-G).

A portion of messages are termed a **corpus**, and the complete dataset from a group (stored as multiple corpora) is called a **capture** (see Appendix G for technical implementation).

The **thread** of a typical post on a social network (such as Facebook, see Figure 6.1) is structured:

1. A user posts a **message**.

2. Users (including the first user) can post a **reply**.

3. Users can **like** the message and/or replies including their own posts.

A message or reply is equivalent to an annotation (A), whilst a 'like' is equivalent to an agreement validation (V_a), as previously defined (see Section 5.1.2).

6.3 Data

In order to investigate inherent problem solving on social networks, several social network (Facebook) groups were selected as they were thought likely to contain good examples. These groups were identified using the inbuilt search functionality on the social network, group recommendations and checking the group membership of prominent users in groups already found. Only groups that were sufficiently mature were selected[1] and were categorised according to purpose and generality (see Section 6.2).[2] The total cached message database includes 34 groups from Facebook containing 39,039 threads and a total of 213,838 messages and replies. The data were transformed into an anonymous database so users cannot directly be associated with the data stored. This use of data is in line with Facebook's Data Use Policy.[3]

Images were not cached, but for the investigation into quality it was necessary to store some images locally in order for them to be manually annotated without bias. All source and copyright information was stored in the database along with an image identifier.

Finding message threads likely to contain the task Messages posted to a group on Facebook can be one of six types: photo; link (URL); video; a question (in the form of an online poll); a scheduled event; or just simply text (status)[4] although the majority of messages are either 'photo', 'link' or 'status' (see Figure 6.2).

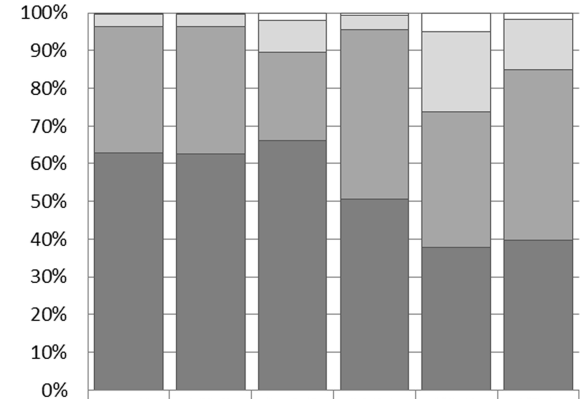
The Task Request (TR) and Media Gallery (MG) groups have more photo type messages posted in them compared to Knowledge Sharing (KS) groups both in the general and topic-specific categories (TR n(6,350) 62.5%, MG n(17,831) 64.2%, KS

[1] Only groups with over 50 messages and 50 members were selected.

[2] The group categorisation was done independently by Jon Chamberlain and two postgraduate researchers at the University of Essex. When there was not consensus on the categorisation (18%), a final decision was made by Jon Chamberlain after group discussion.

[3] `https://www.facebook.com/full_data_use_policy` (15/11/2013)

[4] `http://fbrep.com//SMB/Page_Post_Best_Practices.pdf`



	TR-G	TR-S	MG-G	MG-S	KS-G	KS-S
Other	5	22	302	16	681	20
Link	45	160	1331	82	2901	154
Status	471	1676	3655	989	4925	526
Photo	881	3090	10341	1115	5192	459
	1402	4948	15629	2202	13699	1159

Figure 6.2: Distribution of thread types by group category.

n(14,858) 38.0%, p<0.01, z-test). This is not surprising as the primary motivation for posting a message in TR and MG groups (seeking an identification or showing off a picture, respectively) requires an image to be attached. The KS groups show a more even distribution of message types as motivations for posting (arranging meetings, sharing research, posting information, etc.) do not require an image. This makes TR and MG groups better places to look for image classification tasks.

These messages were cached for analysis (see Appendix G); however, a full scale system would require a larger enterprise solution.

6.4 Annotation scheme

Problem solving on social networks, much like Community Question Answering (cQA), occurs through the natural language of the message thread.

For the purposes of this research, messages and replies were categorised by **inquisition** (question or statement) and **data load** (a solution to the task, see Table 6.1), although more detailed schemas [Bunt et al., 2012] and richer feature sets [Agichtein et al., 2008] have been used to describe cQA dialogue. The message and its replies form a thread that relates to what has been posted (photo, link, etc.). The thread may

Table 6.1: Categories of posts with examples of content, conditional on inquisition (question or statement) and data load (in this case the scientific name of a species in the image).

Category	Content
QUESTION	What is this?
CHECK	Is this Chromodoris magnifica?
NEUTRAL	Great photo from the trip!
ASSERTION	This is Chromodoris magnifica

Table 6.2: Categories of threads when viewed as a task with solutions.

Category	Message	Reply
None	NEUTRAL	NEUTRAL
Unresolved	NEUTRAL	QUESTION
	QUESTION	QUESTION or NEUTRAL
Implied	NEUTRAL	CHECK or ASSERTION
	ASSERTION	Any
Suggestion	CHECK	Any
Resolved	QUESTION	CHECK or ASSERTION

contain solutions (or related data) to tasks, irrespective of whether the poster posed a question in the original message, as other users might augment or correct the posts (see Table 6.2).

6.5 Social learning

Users within groups typically learn how to interact with each other and how to post questions and replies by observation of the group's message feed. Administrators of the group set the rules of engagement in a short description of the group or with a pinned post, as well as advising members directly. These rules tend to proliferate across the group so over time the administrative load is reduced and the members become self-regulating.

As an example, a common explicit guideline within marine species identification groups is to specify the location where the image was taken as this may have an

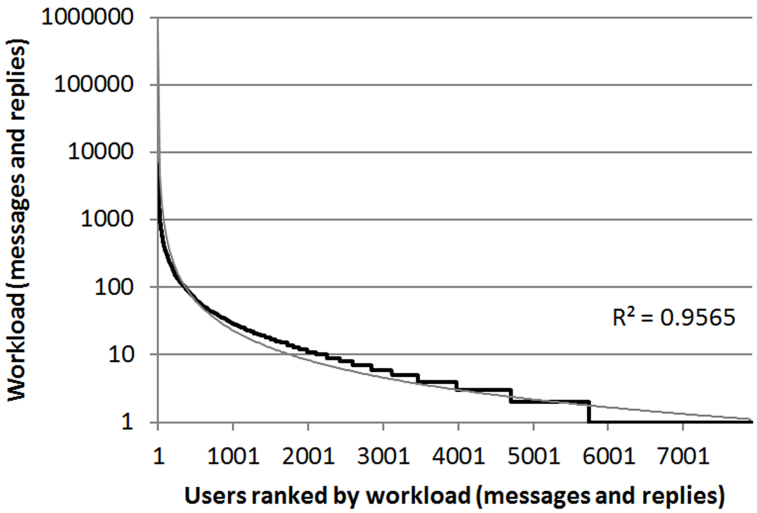


Figure 6.3: Chart showing the workload (messages and replies) of users of the Facebook groups, ranked by total workload.

important bearing on what the species might be (some marine species have limited geographical distribution patterns).[1]

Social learning, in which users on the social network teach and support each other in an ad-hoc manner, encourages users to engage in the learning process to an extent that suits their interests and time constraints. Some users will learn enough to be able to answer other users' questions reducing the traditional bottleneck of a few experts having to do the majority of the work. The annotation scheme is typically the first thing the users learn through social learning.

6.6 Data analysis

Analysis of a random sample of 1,000 messages from the corpus showed a rapid drop in replies to messages after four weeks. Therefore, for the purposes of analysing thread activity, all messages less than eight weeks old from the date of capture were ignored to reduce any bias in message activity of newly-posted and currently-active messages.

[1]With other social media sharing sites such as Instagram and Flickr the image may be automatically geotagged in the EXIF data; however, with underwater images this is often not the case.

Table 6.3: A table summarising groups' (mean) active users (a user who has posted a message or reply) and the median and mean workrate (messages/replies per active user).

	Active users	Workrate (median)	Workrate (mean)
TR-G	28.0%	4	20.8
TR-S	36.5%	4	22.4
MG-G	20.3%	3	12.8
MG-S	32.4%	4	14.5
KS-G	18.4%	3	20.9
KS-S	38.3%	4	11.4

6.6.1 User workload

Collaborative systems, in which workload is shared without control, frequently see a Zipfian distribution of workload with only a small proportion of the users doing most of the work (see Section 2.3.1).

The workload of each user who was a member of the groups analysed was calculated as a total of all messages and replies they had posted. The users were then ranked by workload and, as expected, this follows a Zipf power law distribution (R^2=0.957, see Figure 6.3). The distribution does not show unusual behaviour of the low-workload users as was seen with the *Phrase Detectives* interfaces (see Section 4.7.1), perhaps due to the social nature of the training.

In addition we find that the top 1% of users (n=79) have contributed 41.6% of the work, the top 10% of users (n=792) have contributed 79.2% of the work and the top 20% of users (n=1,583) have contributed 88.4% of the work. This is a more unevenly-distributed workload than the Pareto Principle would suggest (that 20% of the users do 80% of the work (see Section 2.3.1); however, 53.5% of the 14,793 users who were members of the groups had contributed some form of work, much higher than the 1% rule would suggest (that 90% of all users will not contribute anything).

The implications are that whilst the workload is unevenly distributed, social networks have an active membership, perhaps because the barriers to contribution are lower than in other crowdsourcing systems (see Section 3.1.3).

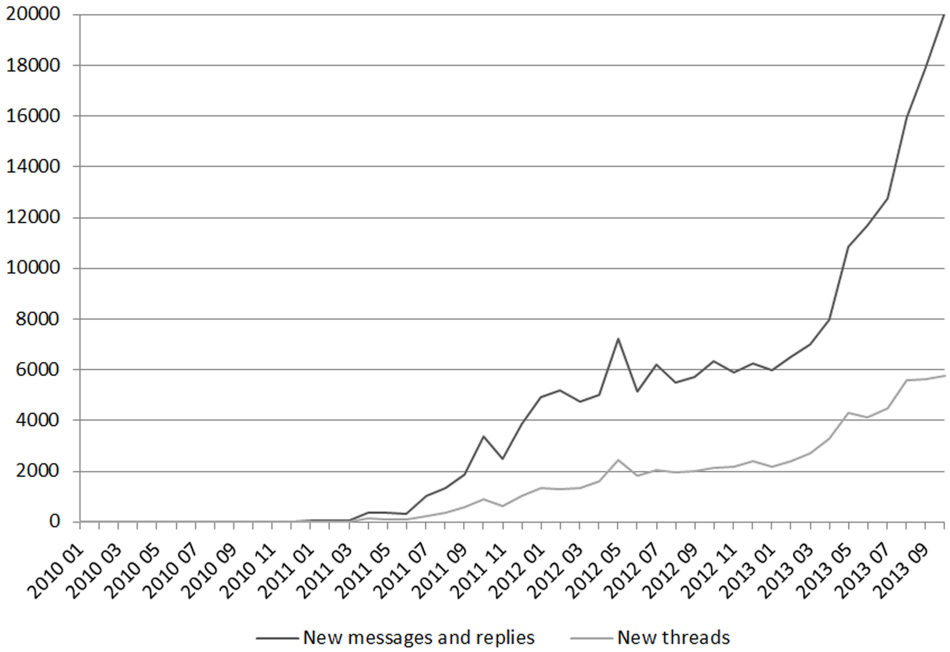


Figure 6.4: Chart showing the amount of new threads and new messages/replies being added to Facebook groups each month.

6.6.2 User activity

User activity was calculated as the proportion of group members that had posted a message or reply from the total membership at the time of the capture (see Table 6.3).

Topic-specific groups have more active users ($p<0.05$, z-test, see Table 6.3), an indication that the community of users in these groups are more engaged with the subject matter and may even know each other personally (as specialist research areas tend to be quite small).

The TR groups have more active members who perform at a higher workrate ($p<0.05$, z-test, see Table 6.3) than the MG groups, supporting the idea that users joining TR groups are more willing to participate actively in problem solving. Users of MG groups may be more passive by simply enjoying the images being shared.

It is clear that there is a lot of information being added to social networks such as Facebook that could be analysed in this way; however, the exponential rise in new data being added each month (see Figure 6.4) will prevent the use of manual analysis techniques in the long term and will require automation.

Table 6.4: A table summarising group categories: the proportion of messages that received a reply; the number of replies (median and mean); the response time (median) for the first reply (hh:mm:ss); the lifespan (median) of the thread (hh:mm:ss); and the proportion of outlier replies beyond 1092:16:00.

	Received a reply	Replies (med)	Replies (mean)	Response time	Lifespan	Outliers
TR-G	81.5%	3	4.1	00:28:30	16:26:16	2.3%
TR-S	71.0%	2	3.2	00:48:57	11:55:09	1.5%
MG-G	42.7%	0	1.6	00:58:25	10:25:50	1.4%
MG-S	49.4%	0	1.8	01:59:46	16:39:43	4.0%
KS-G	50.5%	1	2.8	00:28:29	07:34:21	0.6%
KS-S	58.5%	1	2.2	01:24:45	18:12:20	3.1%

6.6.3 Thread response time, lifespan and activity

The time to the first response (response time) and time to the last response (lifespan) were plotted on frequency graphs (see Table 6.4 and Figures 6.5 and 6.6). 5-10% of messages receive a reply in eight minutes. The proportion of messages with replies beyond 1092:16:00 (6.5 weeks) from the time of the message being posted (outliers) is small so it makes an appropriate cut-off point for message analysis to ensure that messages have had a chance to receive all replies. The graphs show different profiles, indicating that response time is less predictable than lifespan.

General (-G) groups have a faster response rate and a shorter lifespan than topic-specific (-S) groups for MG and KS ($p<0.05$, unpaired t-test, see Table 6.4) perhaps indicating that users in general groups have a broad interest and make conversational replies that do not require a task to be solved.

Within topic-specific categories, the TR groups have a faster response time and shorter lifespan ($p<0.05$, unpaired t-test, see Table 6.4) perhaps because users of these groups anticipate task requests and are primed to submit a reply, especially if it is an opportunity to demonstrate their knowledge. This would be harder to achieve in general groups because the task posted may be outside the knowledge of most users.

Response time and lifespan are influenced by the interface design of social networks such as Facebook. When messages are first posted they appear on a user's news feed and/or notifications and the group wall. Over time they are replaced with other mes-

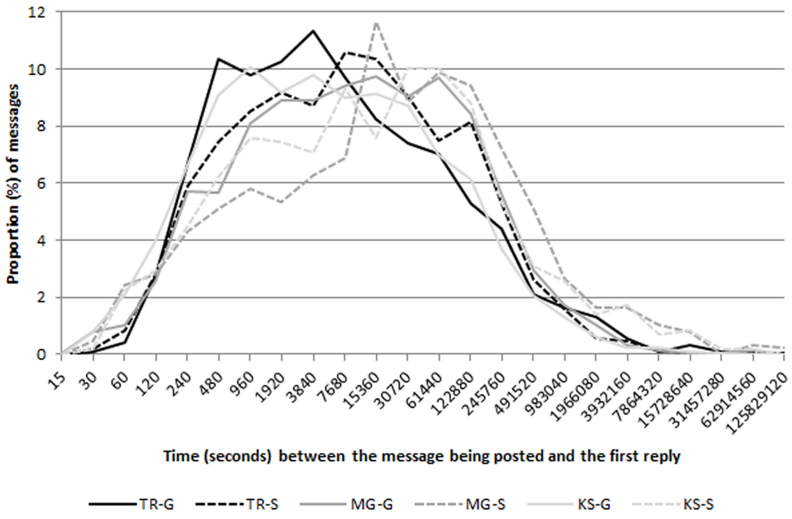


Figure 6.5: Response time (seconds, log scaled) for a thread.

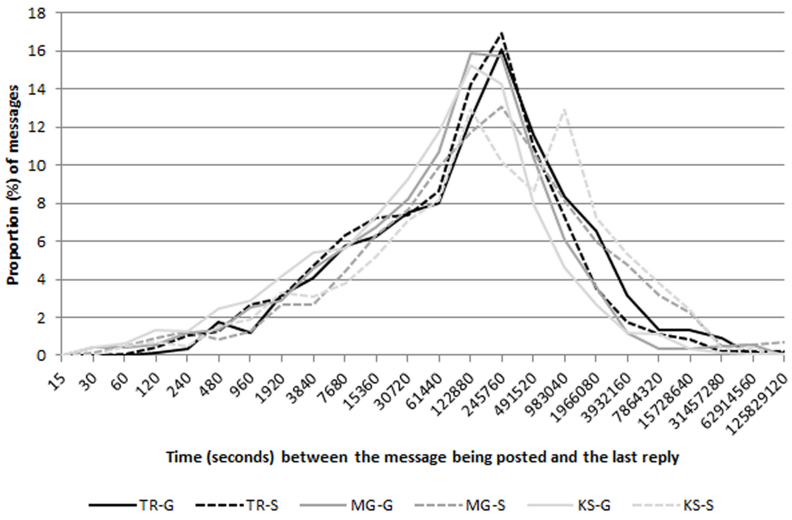


Figure 6.6: Lifespan (seconds, log scaled) of a thread.

sages, move down the page until no longer visible and can only be accessed by clicking for older pages. If a message receives a reply it is moved back to the top of the page (termed 'bumping').

Messages posted in the TR groups have more replies than the other groups ($p<0.05$, unpaired t-test, see Table 6.4). This is unsurprising as these groups are used for posting tasks that require a response, unlike the more passive nature of other groups. This makes the TR groups a good candidate for collective intelligence because more users are potentially involved in the solution of the task.

6.7 Data quality

In the same way *Phrase Detectives* looked at different levels of granularity of data, so too does the work on social networks. Marine species are organised in a hierarchical taxonomy from the broadest levels (phylum to genus) down to the most specific level (species). Species level is actually constructed of two parts: the *genus* which represents several closely-related marine species and the *species epithet*, which distinguishes between closely-related animals. For example, in the species 'Chromodoris magnifica', *Chromodoris* is the genus name and *magnifica* is the species epithet.

In this research we look at species level annotations because identification through morphology is more precise and easier to determine if correct or not.[1]

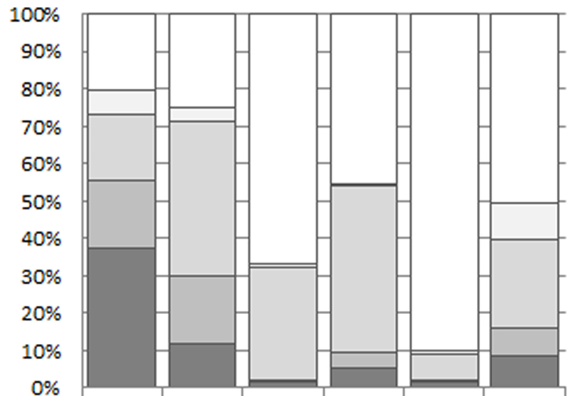
6.7.1 Task distribution

In order to assess the quality of data that could be extracted, and to investigate the distribution of the tasks within the group categories, 200 threads were selected at random from each category to form a subcorpus of 1,200 threads.

The subcorpus was manually categorised in a random order for data load and inquisition (see Section 6.4) by only viewing the thread text and author names, thus each thread could be classified as a task type (see Table 6.2).

Implied, Suggestion and Resolved tasks all contain data that could be extracted to solve the image classification tasks. TR groups have more data-loaded threads than

[1] This reasoning is likely to be questionable to some taxonomists who prefer species to be determined by DNA sequencing rather than morphology; however, it is accepted practice to identify animals to species level using physical characteristics.



	TR-G	TR-S	MG-G	MG-S	KS-G	KS-S
None	41	50	134	91	180	101
Unresolved	13	7	1	1	2	20
Implied	35	83	61	89	14	47
Suggestion	36	36	1	8	1	15
Resolved	75	24	3	11	3	17

Figure 6.7: Distribution of image classification tasks by group category.

MG or KS groups ($p<0.05$, z-test) and it is not surprising due to the purpose of the groups (see Figure 6.7). Additionally, tasks are more likely to be solved in the TR groups comparing resolved tasks to unresolved tasks ($p<0.05$, z-test).

6.7.2 Baseline measures

Based on the previous findings it could be expected that the highest frequency of task requests and more accurate solutions would be found in the TR-S groups, although there are fewer explicit tasks compared to TR-G. A single topic-specific area of Opistobranchia (sea slugs or nudibranchs) was chosen in order to evaluate the accuracy of image classification. In this class of animals external morphology is often sufficient to confirm a classification from an image (unlike, for example, marine sponges) and this is also an active area on social media.

A random sample of threads from two groups (Nudibase[1] and NE Atlantic Nudibranchs[2]) from the TR-S subcorpus was taken. Only photo threads were selected and further threads removed if they were unsuitable for the image classification task (for

[1] `https://www.facebook.com/groups/206426176075326`
[2] `https://www.facebook.com/groups/NE.Atlantic.nudibranchs`

example, not an Opistobranch, multiple species in an image, close-ups, words printed in the image, continuation and/or gallery threads).

In total 61 threads were manually analysed using this method (called the **test set**).

The gold standard, created by examining eight resources (see Appendix H), was compared for inter-expert agreement using Fleiss' kappa, which allows for more than two annotators (unlike Cohen's kappa). This test showed an inter-annotator agreement of κ=0.61, considered to be substantial agreement (n(84), raters(8), z=34.1, κ=0.61, Fleiss' kappa). This is perhaps an underestimation of agreement between the resources as it accounts for all the images in the test set, including those when no classification was found.

By way of comparison the two resources that produced the most classifications (*Seaslug Forum* and *Nudipixel*) have very high agreement (n(84), raters(2), z=9.2, κ=0.84, Cohen's kappa), more in line with what could be expected from expert annotators in linguistic settings (see Section 5.1.1). Additionally, these two resources only disagreed with the classification on one occasion giving an inter-annotator agreement accuracy of 98.3% (counting only instances when both resources had a classification) which could be considered the top performance expected from any automatic aggregation of the groupsourcing data.

By using the gold standard to determine which answer from the subcorpus was correct, results show very high accuracy for the image classification task (0.93), see Table 6.6. This represents the upper limit of what could be expected from groupsourcing as other categories of groups may have lower performance.

Filtering to improve quality User workload, rating and response time were not effective methods of filtering data in *Phrase Detectives* so were not investigated here. System errors do not exist in this dataset in the same way as they did in *Phrase Detectives* so were not implemented as filters.

There were considerable numbers of posts that did not contain information and these can be safely ignored, although they would also ideally be prevented from entering the system to reduce overhead.

Table 6.5: A table summarising the scores of correct and incorrect answers using either messages and replies (A) or messages, replies and likes ($A + V_a$).

	N	Mean	SD	Min	Med	Max	Intersect	Overlap
A							1.1	69.8%
Gold standard	55	1.27	0.83	1	1	6		
Incorrect	16	0.69	0.7	-1	1	1		
$A + V_a$							3.7	68.4%
Gold standard	55	4.85	4.62	-10	4	21		
Incorrect	16	1.38	4.05	-12	2	6		

6.8 Aggregation using the AV Model

There is a difference between the mean scores of correct and incorrect answers, both when using cumulative messages and replies (A, p=0.012, unpaired t-test) and messages, replies and likes ($A + V_a$, p<0.01, unpaired t-test), see Table 6.5. However, it is unclear whether the AV Model could be used automatically to determine correct answers with confidence due to the large overlap size.

Additionally, there were very few negative statements (0.14 mean negative statements per thread) which could be an indication that a 'disagree' button (disagreement validation V_d) would not be used as much as a 'like' (agreement validation V_a) button.

6.9 Comparison to microworking

The images from the subcorpus (called the **test set**, defined further in Section 6.7.2) were also classified using Crowdflower[1] to compare the accuracy. Crowdflower users were presented with an image and asked to provide a species name (see Figure 6.8). Web resources were mentioned in the instructions, as well as the requirement for accurate spelling although minor capitalisation mistakes and synonyms were allowed (see Figure 6.9). The Crowdflower configuration selected the top 36% of users on the system to work on the task who were offered $0.05 per image annotated, with ten answers required for each image.

[1] `http://www.crowdflower.com`

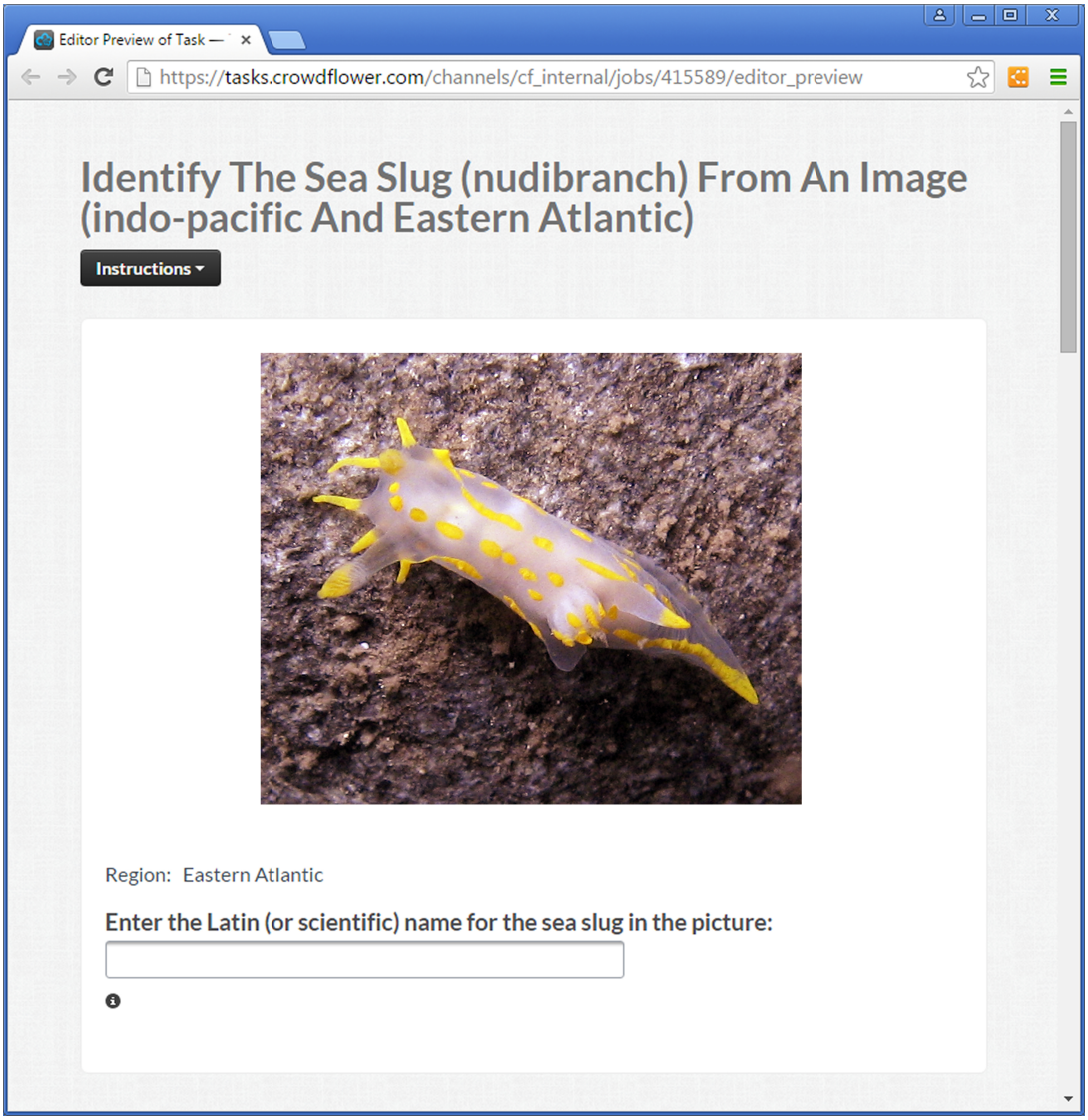


Figure 6.8: Screenshot of the Crowdflower task.

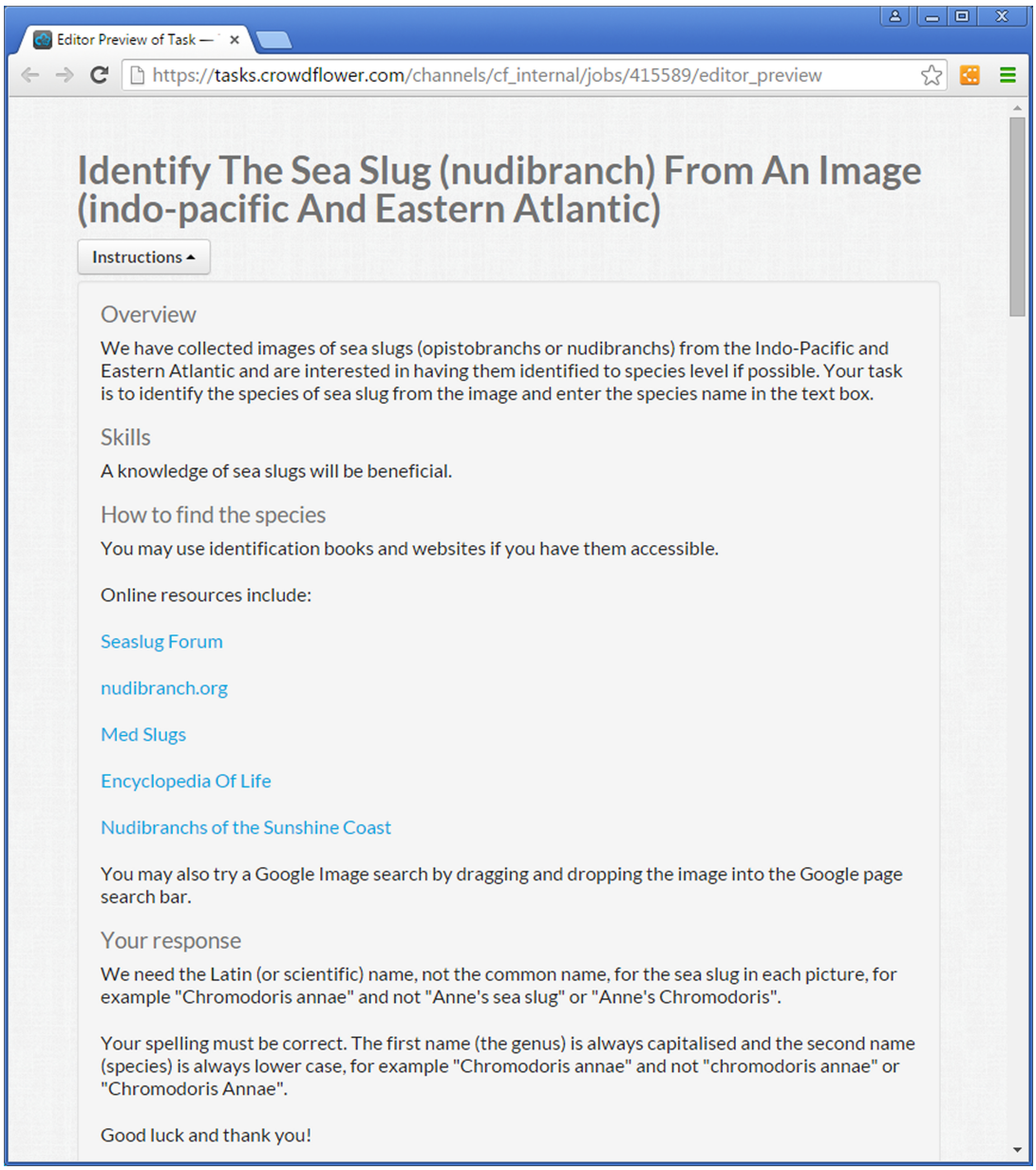


Figure 6.9: Instructions screen for the Crowdflower task.

Table 6.6: Comparison of image classification accuracy between different crowdsourcing methods.

Crowdsourcing method	Accuracy
Inter-expert (test set)	0.98
Groupsourcing (test set)	0.93
Crowdflower (training) @ $0.05 n=10	0.91
Crowdflower (test set) @ $0.05 n=10	0.49

A **training set** of 20 images with known answers was created with the most common sea slugs found on the photo sharing website Flickr.[1] This dataset was used both as a training gold standard (i.e. the users were told if their answers agreed with the known answer) and also as a benchmark annotation dataset. Users were presented with images from both datasets, with high-performing (according to Crowdflower's assessment of performance against the gold standard) users' data being labelled as 'trusted'. In total 1,525 annotations were made, from 72 users, of which 701 annotations were considered 'trusted' by Crowdflower. The data collection cost $104. Users rated (out of five):

- the task instructions (3.4);
- the fairness of the question (3.0);
- the ease of the task (2.0);
- the pay (3.3).

Results show that with microworking there was high accuracy in the training set, but the test set scored much lower accuracy (see Table 6.6). This is an indication of how hard the task was in the test set and if task difficulty is extrapolated to groupsourcing it would achieve an accuracy of 0.99 on the training dataset.

6.10 Summary

In this chapter, social networks were explored to see whether they contain examples of problem solving, to what extent they contain good answers to those problems and

[1] `https://www.flickr.com`

to gain an idea of how difficult it would be to extract that information automatically, both by analysing the conversation associated with the image and understanding how to aggregate the group responses including posts and likes.

In order to answer these questions a corpus of messages was extracted, including 34 groups from Facebook containing 39,039 threads and a total of 213,838 messages and replies. As expected, users of all groups distribute work unevenly (the top 20% of users do 88.4% of the work), typically following a Zipf distribution.

Groups that are set up specifically for users to post and solve problems show the most promise for collective intelligence, with users having a higher workrate, faster response time, shorter message lifespan and more in-thread activity and discussion. Problems posed in these groups are likely to get a faster reply, find an answer faster, elicit more data from users and more likely to have the task completed.

Tasks posed in such groups tend to be difficult; however, the quality is very high when compared to experts and when compared to a microworking approach.

There is a clear difference between the scores of correct and incorrect solutions using the AV Model; however, a larger study is required before understanding if this process can be done automatically. Automatic processing of these types of data are essential given the rate of increase of data being added every day to social networks.

7

Discussion

The theoretical and experimental work of this thesis has covered the different aspects of the primary question as to whether social networks can be used to create large-scale data resources, with high-quality labelling of information about the data. This chapter discusses the work in the context of the existing research landscape, the limitations and whether this approach can be used to create knowledge to solve problems that cannot at present be addressed in any other way.

7.1 Data acquisition and annotation

Crowdsourcing approaches are typically used by a requester who has data they would like a task performed on; however, it may also be the case that the requester can acquire the data as part of the task, as seen in citizen science approaches, or even align with existing efforts, as has been seen with groupsourcing. It ultimately becomes a question of scalability: in order to scale up efforts for collecting large resources for machine learning, every conventional bottleneck must be removed and this is why social networks are so appealing. Users are motivated to answer the same type of questions as the requester and moderate themselves to ensure the resource is of high quality. Directing such a community of users is not straightforward and attempts at central control may give rise to resentment from some quarters. This makes the groupsourcing approach difficult when there is a shortage of skills or little general interest in the wider community.

Portions of this chapter previously appeared in Chamberlain and O'Reilly [2014]; Chamberlain et al. [2013]; Chamberlain [2014a,b,c]; Poesio et al. [2013].

When groups of users can be found creating data and performing tasks on them, the tasks are likely to be getting a faster reply, find an answer faster, elicit more data from users and are more likely to have the task completed. It may also be the case that the task is being performed implicitly by the users and this may reveal an additional wealth of high-quality data.

Coupled with the bias of what data users want to work on, is the issue of data sparsity in general. In the context of anaphora there may only be few examples of genuine ambiguity that are of real interest in a document set. In a similar way with the citizen science project *GalaxyZoo*, only a few rare instances of unusual features are of real interest, although the general classification work assists with creating a large resource. In the context of images of marine species, some animals are more charismatic and easy to find than others, or are physically more common and well distributed, therefore there will be more images of these posted on social networks. Again, it is the discovery of rare incidents of unusual data that is of most interest and discerning these outliers from mistakes or malicious input is a major challenge for an autonomous system.

Allowing participants in scientific activities a wider range of input may be the key to knowledge discovery and this is a serious shortcoming of human computation and the games-with-a-purpose approach. By working on pre-selected data and restricting the input of the users, one may not be able to maximise the ability of humans to perform complex tasks. An unconstrained approach such as peer production allows the data to evolve in a way that interests the community, for example in the case of marine life, annotating interactions with other species, population dynamics, geographic distribution and other niche dimensions that could be indicators of ecosystem changes caused by pollution, overfishing or climate change.

A groupsourcing approach challenges what is known about a topic to cast a more realistic (although likely to be biased) view. This relates to the idea of a functional niche (i.e. the maximum parameters under which the concept as a whole could exist) compared to the realised niche (i.e. under what parameters individuals of the concept have been observed) [Hutchinson, 1957]. For example, a marine species may have a thermal tolerance such that it could theoretically survive in cold Arctic waters (its functional niche), but has never been observed in such waters (its realised

niche). Similarly, ambiguous anaphora may theoretically exist, but are never observed in documents.

Crowdsourcing approaches typically require some form of pre-processing to get the data ready for the participants and post-processing to clean up the submitted annotations. In *Phrase Detectives*, the documents were manually selected and prepared before being converted by a pre-processing pipeline that extracted markables from the text (see Appendix C). This process was time-consuming and many errors were introduced into the system that had to be corrected later by an administrator. The data from social networks were collected after the users had created it and added their annotations; however, some processing was required to remove unsuitable data before they could be converted into a usable corpus (see Appendix G). Given the need for truly large-scale resources the pre-processing stage of the data needs to be as high-performance as possible because errors have a considerable knock-on effect through the system.

One final point about the data is the way they are structured. Hierarchical structures for organising knowledge can be unstable, for example, the taxonomy and identifying morphology of marine species is in constant flux, meaning identifications previously considered correct may have changed. There was a significant update to the taxonomic group *Chromodorididae* [Johnson and Gosliner, 2012] that rendered many static Web resources and books out of date; however, users frequently correct identifications to the new nomenclature on social networks.

7.2 User motivation

Crowds can be motivated in different ways, dependent on the system, task and goals, but overall the success of incentives can be measured by how much the people participate and how much they contribute.

In terms of player recruitment, the standalone *Phrase Detectives* game was more successful than the version embedded on a social network; however, the latter system had a higher conversion rate of casual users to trained players capable of contributing useful work (26.2% compared to 6.6%). This is an important point, because whilst it is useful to attract many people to a crowdsourcing effort, that crowd needs to do some work. The level of recruitment in *Phrase Detectives*, whilst not in the same league as the *ESP Game* which enjoyed massive recruitment in its first few months online, could

be seen as what you would expect if some effort were made to advertise a GWAP and motivate people to play it. The conversion of casual users to trained players in a similar language game on Facebook showed a similar conversion rate (24.3%), indicating that this could be the norm [Herdagdelen and Baroni, 2012].

Users that contribute nothing are consuming site resources (from bandwidth to interaction with administrators), as well as potentially producing spam or malicious content if they can access the system without a training stage. It may also be the case that users who complete a training stage just try out the system and give up very quickly; however, these players contribute so little data to the system that they are not worth filtering out.

Non-contributing members of a collective effort are commonplace on social networks, with most users simply viewing the content rather than contributing to new content or commenting on existing content. Social networks have a very low barrier for participation, in that a user can simply 'like' content rather write a comment.

Motivating contributions Participation (or volition) of users to contribute is a way to assess whether the incentives of an approach are effective. An active user is described as one who contributes some work during a specified timescale. The standalone version of *Phrase Detectives* had more active players than the Facebook version; however, the latter version's players did more work per player.

Another way to view contribution to a system is how much time each user contributes. The average weekly contribution for Wikipedia is just over eight hours [Nov, 2007]; however, this is for contributing users of Wikipedia, not for casual browsers of the website. This indicates that when a user starts contributing to Wikipedia they are highly motivated to contribute. In Mechanical Turk the contribution rate is a little lower, between four to six hours [Ipeirotis, 2010b], and it can also be expected that the user, once registered, will be highly motivated to contribute.

There is a huge complexity and spread of user types within the Mechanical Turk user base; however, it is interesting to note that for 20% of the workers, this represents their primary source of income (and for 50%, their secondary source of income), and they are responsible for completing more than one third of all the HITs [Ipeirotis, 2010a]. Participating for leisure is important for only 30% of workers so the motivations for participating to microworking are very different from that of Wikipedia.

An observation in most crowdsourcing systems is the uneven distribution of contribution per person, often following a Zipfian power law curve. This was certainly the case for the *Phrase Detectives* game and for the social network groups that were investigated in Chapter 6. Similarly, studies of microworking also find that only 20% of the users are doing 80% of the work [Deneme, 2009].

On social networks, groups that are set up specifically for users to post and solve problems have users working at a higher rate with more in-thread activity and discussion than more general groups, an indication that groupsourced tasks is inherently motivating for the community, although a wider study would be required to generalise this finding.

Altruism in the community Crowdsourcing may initially attract collaborators by giving them the sense that they are contributing to a resource from which everyone may benefit and these are usually the people that will be informed first about the research. However, in the long term, most of the participants of crowdsourcing will never directly benefit from the resources being created. It is therefore essential to provide some more generic way of expressing the benefit to the crowd, i.e. the value of what the requester is doing.

For example, this was done with *Phrase Detectives* in a BBC radio interview by giving examples of natural language processing techniques used for Web searching. Although this is not a direct result of the language resources being created by this particular project, it is the case for efforts of the community as a whole, and this is what the general public can understand and be motivated by.

This purpose to data collection, common also in citizen science and peer production approaches, has an advantage over microworking, in which the workers are not connected to the requester. There is a sense of ownership, participation in science, and generally doing something useful. When users become more interested in the purpose of the crowdsourcing rather than the system itself it becomes more like a citizen science approach in which users voluntarily work on harder tasks, provide higher quality data and contribute more.

The indirect financial incentives in *Phrase Detectives* showed a strong correlation with generating more work from the players. This is an intuitive assumption, but maximising how rewards are distributed to get value for money will always be an issue,

as well as ensuring participants do not feel cheated if they do not receive an award when they believe they are entitled to one.

7.3 Group homogeneity

It has been shown that moderately-diverse groups are better at solving tasks and have higher collective intelligence (termed c) than more homogeneous or very diverse groups. A balanced gender ratio within a group also produces a higher c as females demonstrate higher social sensitivity towards group diversity and divergent discussion [Woolley et al., 2010].

Gender distribution in crowdsourcing approaches can be varied. Demographics of *Phrase Detectives* players [Chamberlain, Kruschwitz, and Poesio, 2012] support previous surveys that show women are more likely to play, and will spend more time playing, online games, especially if linked to social networks.

Facebook generally is also reported to have more female users[1] although, in the case of the social network groups investigated, there was a clear bias towards male users [Chamberlain, 2014b]. Similarly, only 12% of contributors to Wikipedia are female [Glott, Schmidt, and Ghosh, 2010], a statistic that prompted significant research into the gender bias in the authorship of the site [Laniado et al., 2012]. It may be that groupsourcing is appealing in the same way as Wikipedia, or perhaps males prefer image-based tasks to word-based problems to solve [Mason and Watts, 2009], or even that the topic is a male-dominated interest (66% of PADI diving certifications in 2010 were for men).[2]

A survey of microworking site Mechanical Turk workers initially showed a similar gender divide in participants when the system was mainly populated by US workers (65% female) [Ipeirotis, 2010b]. More recent surveys showed that the changing demographics of the workers, driven by allowing payment to Indian workers in rupees, now have more male workers from India who use microworking as a primary source of income [Ross et al., 2010] and the gender ratio is almost even [Ipeirotis, 2010b].

The changing demographics of crowdsourcing participants may have an impact on the types of incentives and tasks offered, as well as the overall quality from the system

[1] http://royal.pingdom.com/2009/11/27/study-males-vs-females-in-social-networks
[2] http://www.padi.com/scuba-diving/about-padi/statistics/pdf

due to the group's homogeneity, although systems in which the users work collectively will be less affected as there is no direct contact.

7.4 System throughput

A measure of efficiency of the interface and task design is how fast tasks are being completed or annotations generated. This measure is called **throughput**, the number of labels (or annotations) per hour [von Ahn and Dabbish, 2008]. The throughput of *Phrase Detectives* is 450 annotations per human hour, which is almost twice as fast as the throughput of 233 labels per human hour reported for the *ESP Game*. There is a crucial difference between the two games: *Phrase Detectives* only requires clicks on pre-selected markables, whereas the *ESP Game* requires the user to type in the labels, which highlights the importance of the interface design.

The throughput of Mechanical Turk has been reported to be close to real time (within 500ms of a HIT being posted) but this is usually for very simple tasks [Bigham et al., 2010]. More complex tasks can take up to a minute to complete giving a throughput range from one to 7,200 labels per hour and some may never be completed. Whilst these figures are not especially helpful, it highlights the potential speed of this approach if the task can be presented in an efficient way.

Designers of crowdsourcing systems who are considering making their task timed should consider the speed at which the user can process the input source (e.g. text, images) and deliver their response (e.g. a click, typing) in order to maximize throughput and hence the amount of data that are collected.

Related to throughput is the **wait time** for tasks to be done. Crowdsourcing systems that allow data collection in parallel (i.e. many participants can work at once on the same tasks) are the most effective at dealing with the wait for a user to attend to the task. Such systems can have multiple tasks live on a system for users to work on. Although the throughput may give us a maximum speed from a system, it is worth bearing in mind that the additional time spent waiting for a user to be available to work on a task may slow the system considerably.

This is when the microworking approach, with a large worker pool, has an advantage and some task requesters even pay workers a retainer to be on demand [Bernstein et al., 2012]. With other approaches it is possible to prioritise tasks to maximise completion

of annotation, but for open collaboration such as Wikipedia and social networks it is much more difficult to direct users to areas that need contribution. This can be seen by comparing popular pages that have considerable work, such as for the film *Iron Man*[1] with 8,000 words, with less popular pages, such as *Welsh poetry*[2] with only 300 words.

The combination of throughput and wait time make microworking an attractive option for completing tasks with a crowd as the time to complete a job is more predictable and, if money were no object, would clearly be the fastest approach.

The idea of throughput does not naturally translate to social networks because of the way tasks are interacted with. From the analysis here we know that the vast majority of threads do not have a reply after 6.5 weeks; however, that is not the same as determining when the task is complete and masks the subjective observation that images, in particular ones that are easy to identify, are classified within minutes. A different way to look at the throughput might be the maximum threads that could be realistically posted to a group per hour to give the community enough time to respond to them. Presumably there will be saturation point when too many messages flood a group's feed and it becomes unmanageable. The results of this research show that groups that are set up specifically for users to post and solve problems have a faster response time and shorter message lifespan, implying their throughput and wait time is lower than more general groups on social networks.

Response time as a performance indicator When attempting to analyse and improve a system interface it is often the performance of users that measures the success of different iterations of design. The metric of performance depends on the context of the task and what is considered the most important outputs by the system owners, for example, one system may desire high-quality output from users, whereas another might want fast output from users [Radlinski and Craswell, 2010].

Using response time as a performance indicator presents a different set of problems in that it may not be assumed that speed correlates to quality. Ideally a fast response indicates a highly-trained user responding to a simple task and conversely a slow response indicates a difficult task that requires more thought.

[1] `http://en.wikipedia.org/wiki/Iron_Man`
[2] `http://en.wikipedia.org/wiki/Welsh_poetry`

By understanding the way users interact with a system, each task response time can be predicted. In the case of the *Phrase Detectives* game we can use a prediction of what the user should do for a given size of input to process, task difficulty and data-entry mode [Chamberlain and O'Reilly, 2014]. The same could be applied to any task-driven system, such as search, in which the system returns a set of results from a query of known complexity with a set of actionable areas that allow a response to be predicted even when the user is unknown.

When the system is able to predict a response time for a given input, task and interface combination user performance can be measured, with users that perform as predicted being used as a pseudo-gold standard so the system can learn from new data. Outlier data can be filtered; a response that is too fast may indicate the user is clicking randomly or that it is an automated or spam response; a response that is too slow may indicate the user is distracted, fatigued or does not understand the task and therefore the quality of their judgement is likely to be poor.

Results from filtering the *Phrase Detectives* data on response time indicate that factors other than the user's performance will account for the response time, such as task difficulty. A more precise model could be achieved with eye-tracking and GOMS (Goals, Operators, Methods, and Selection) rule modelling [Card, Newell, and Moran, 1983] using a test group to establish baselines for comparison to the log data or by using implicit user feedback from more detailed logs [Agichtein, Brill, and Dumais, 2006]. Without using more precise measures of response time this method is most usefully employed as a way to detect and filter spam and very poor responses, rather than as a way to evaluate and predict user performance.

Modelling the system and measuring user performance allows designers to benchmark proposed changes to see if they have the desired effect, either an improvement in user performance or a negligible detriment when, for example, monetising an interface by adding more advertising. Sensory and motor actions in the system can be improved by changes to the interface, for example, increasing the contrast or size of the text to allow faster processing of the input text. Decision making can be improved through user training, either explicitly with instructions and training examples or implicitly by following interface design conventions so the user is pre-trained in how the system will work.

7.5 Interface design

The design of the interface will determine how successfully the user can contribute data to a crowdsourcing system. In *Phrase Detectives* the player is constrained to a set of predefined options to make annotations, with freetext comments allowed (although this is not the usual mode of interaction with the game). The pre-processing of text allows the interface to be constrained in this way, but is subject to errors in pre-processing that must also be fixed.

The interface of microworking is also predefined and presents limitations that constitute an important issue for some tasks, for example, in annotating noun compound relations using a large taxonomy [Tratz and Hovy, 2010]. In a word sense disambiguation task, considerable redesigns were required to get satisfactory results [Hong and Baker, 2011]. These examples show how difficult it is to design language tasks for crowdsourcing within a predefined system.

An attempt was made to emulate the anaphoric coreference task in *Phrase Detectives* using microworking; however, this proved to be very difficult as the the users were restricted to entering an imprecise text notation, for example having to write *DO line 2 "the door"* for a highlighted markable or using two inputs to select the class of relation and the where the antecedent is (see Figure 7.1). Given the additional difficulties of pre-formatting the text as an image, this experiment was abandoned in favour of more promising directions. This method also highlighted some of the difficulties of using a groupsourcing approach for language tasks, discussed further in Section 7.10.

The interface design also has an impact on the speed at which players can complete tasks, with clicking being faster than typing. A design decision to use radio buttons or freetext boxes can have a significant impact on performance [Aker et al., 2012].

Players of *Phrase Detectives* preferred using the interface deployed on the social network Facebook despite the fact the interface responded slower due to having to load the Facebook wrapper around the content of the game.

Errors in the game, such as those that were filtered out in this research, constitute wasted effort and should be dealt with by bug testing the system rather than post-processing. The application of post-processing error filters to the annotation and validation decisions increases quality (accuracy) between 5-13%; however, there is a large amount of work discarded (between 16-55%).

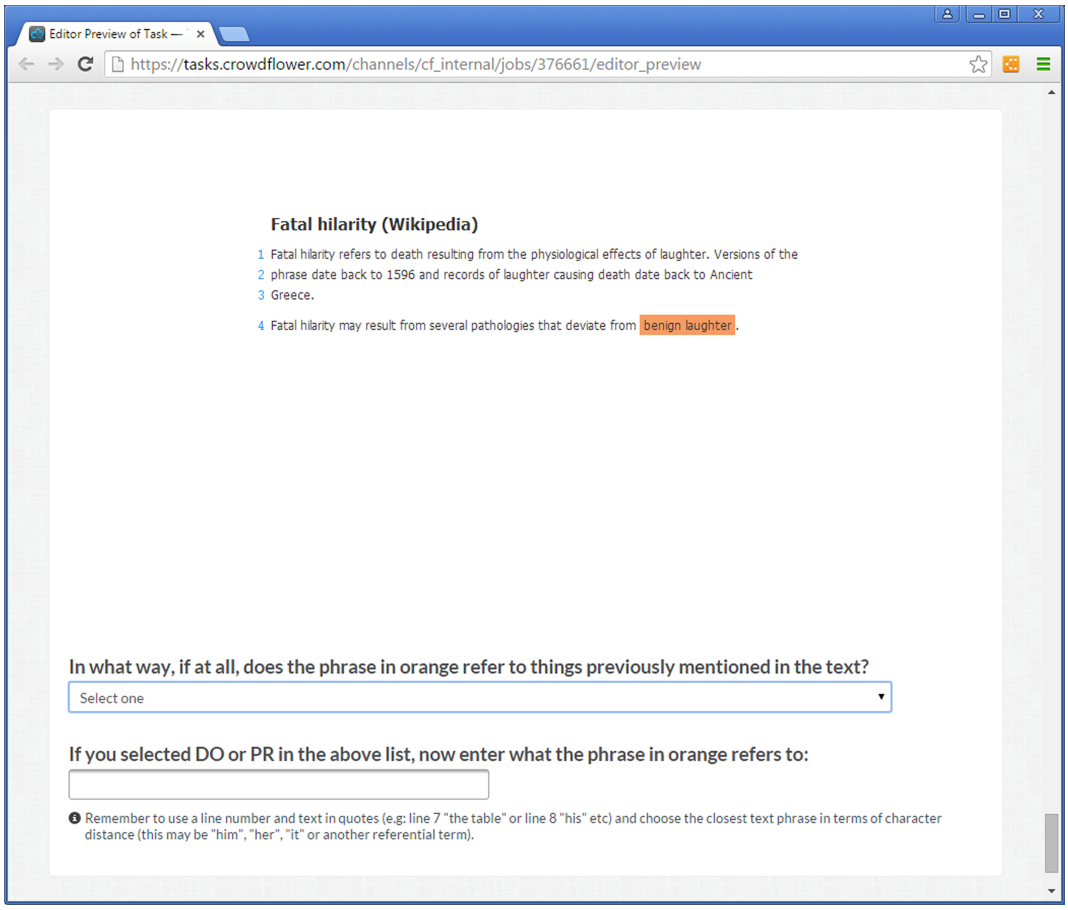


Figure 7.1: Screenshot of the anaphoric coreference task presented in Crowdflower.

The design of social network interfaces is dictated by the owners of the platforms, rather than the requester or the community of users and crowdsourcing efforts may be in conflict with other revenue-generating activities such as advertising.

7.6 Task difficulty

The experimental work of this research has shown that a gaming approach, whether implemented on a social network or not, produces high-quality work from the players, comparable to work of an expert. Additionally, inherent problem solving on social networks can also produce high-quality data if communities of users can be found doing the task. Both the tasks of anaphoric coreference and image classification are not simple and, although the majority of tasks were not hard, it is the uncommon difficult tasks that require the power of human computation. A less-constrained social network environment allows these difficult tasks to be solved in more organic ways compared to the constrained system.

There is a clear difference in quality when we look at the difficulty of the tasks in *Phrase Detectives*. Looking separately at the agreement on each class of markable annotation, we observe near-expert quality for the simple task of identifying discourse-new (DN) markables, whereas discourse-old (DO) markables are more difficult. This demonstrates that quality is not only affected by player motivation and interface design but also by the inherent difficulty of the task. Users need to be motivated to rise to the challenge of difficult tasks and this is when financial incentives may prove to be too expensive on a large scale.

The quality of the work produced by microworking, with appropriate post-processing, seems sufficient to train and evaluate statistical translation or transcription systems [Callison-Burch and Dredze, 2010; Marge, Banerjee, and Rudnicky, 2010]. However, it varies from one task to another according to the defining parameters. Unsurprisingly, workers seem to have difficulty performing complex tasks, such as the evaluation of summarisation systems [Gillick and Liu, 2010].

The community of users in the groups examined on social networks performed image classification tasks at near-expert levels on difficult tasks, considerably outperforming the same set of tasks on the Crowdflower microworking platform. In comparison to

other approaches to wildlife image classification it also outperforms a gaming approach [Prestopnik, Crowston, and Wang, 2014].

Ambiguity A task may be difficult for several reasons: the correct answer is difficult, but not impossible, to determine; the true interpretation is a difficult type of solution to determine; or that the answer is genuinely ambiguous and there is more than one plausible solution. In the Wikipedia and Gutenberg corpora the latter tasks were rare, but are of the most interest to computational linguists and machine learning algorithms. In these cases the users need to have a thorough understanding of how to add their solutions, and this is measured as user credibility, or the chance that the user will select the best answer in line with a gold standard.

Gold standard interpretations in the *Phrase Detectives* corpora have a higher average player rating than incorrect interpretations, with the implication being that player rating can be used as a measure of credibility. Additionally, the players of the Facebook version of the game had higher ratings than the standalone version.

Factors such as document length and readability do not seem to impact quality. However, users do find it harder to detect and annotate different types of interpretation, and the frequency of difficult tasks within different document topics will influence the overall quality obtainable from a system.

The language used on social networks creates even more ambiguity, with ill-formed grammar and spelling, concatenation, contextual referencing and sentiment, for example (taken from the groupsourcing test set):

> 'Is this Coryphella browni or bostoniensis?'
> 'I don't think this is C. brownii.'
> 'I agree with you on that.'

In both the *Phrase Detectives* and groupsourcing corpora the correct gold standard interpretation has a significantly higher confidence score than incorrect interpretations of the same markable so we can have high confidence in answers that score more. There were very few cases of genuine ambiguity in the corpora and automatically processing these cases from the data would be difficult.

7.7 Social learning and the expert in the crowd

One of the distinct advantages of groupsourcing over other crowdsourcing approaches is that the participants learn from each other, not only how to contribute to the system, but also knowledge to solve the tasks. This interaction is led by more experienced and knowledgeable members of the community in an open and transparent way, meaning that when a user receives an answer from an expert, many more may be passively learning from it. Outreach and communicating knowledge to the general public is a core objective of academic institutions and social networks can be used to facilitate these aims. **Social learning**, in which users on the social network teach and support each other in an ad-hoc manner, encourages users to engage in the learning process to an extent that suits their interests and time constraints. There are dangers of convergence towards the opinions of charismatic members or the majority; however, for difficult tasks a degree of discussion and consensus is preferable to majority voting.

The advantage of having an expert in the crowd is that their knowledge is spread through the community and ultimately reduces their workload in the group to only the most unique and difficult cases, which is a primary motivation for the expert to contribute in the first place. Some users will learn enough to be able to answer other users' questions reducing the traditional bottleneck of a few experts having to do the majority of the work. Small groups of annotators will not have the breadth of knowledge required to answer difficult, niche questions [Henry and Roberts, 2014], but a social network community allows experts from other groups to be drafted in.

An issue with all crowdsourcing systems is how to gauge the user's ability to complete tasks, as well as have the internal knowledge required to solve problems. The distinction between a non-expert and expert is often not clear cut [Brabham, 2012b] and prior knowledge may be an important user bias. Additionally, over time human annotators' abilities and biases will change the way they perform tasks which does not make them a consistent, long-term tool [Culverhouse et al., 2003].

The issue of expert bias has also been raised, when collective intelligence systems can be manipluated (intentionally or otherwise) by the perceived ability of an expert to answer a question due to their reputation [Alon et al., 2015]. This is a long-standing issue in research areas of reputation management, expert finding and recommender

systems; however, it is intuitive to believe that the expert in the crowd idea is beneficial to the community.

In addition to experts in the crowd, the idea of crowd-powered experts has also been proposed. A classification task using images of breast cancer showed reasonable accuracy from microworking. By using an approach in which the crowd deal with the majority of the easy work and experts focus on the difficult images, considerable improvements in overall system performance were made [Eickhoff, 2014]. This accuracy is comparable to what could be achieved by groupsourcing and could be considered a similar scenario in which the majority of group users take on the bulk of the work solving easy tasks, leaving the experts to focus on what is of most interest to them.

7.8 Costs of implementing crowdsourcing systems

The goal of crowdsourcing in this research is to create large-scale resources that can be used for machine learning. The traditional method of creating these resources with expert annotators clearly does not scale up in terms of cost, but some crowdsourcing approaches may also not be suitable in terms of projected cost, in particular the microworking approach that uses a per-work reward system. In this section we discuss the actual costs of comparable approaches, including setup time and administration, and whether those costs can be reduced through optimisation of human effort.

When evaluating the costs of the different approaches to collaboratively creating language resources, it is important also to consider other constraints, namely the speed at which data can be produced, the size of the corpus required, and the quality of the final resource. In order to compare the cost-effectiveness we make some generalisations, convert all costs to US$ and calculate an approximate figure for the number of annotations per US$. Where we have factored in wages for software development and maintenance we have used the approximate figure of $54,000 per annum for a post doc research assistant.[1] Additional costs that may be incurred include maintenance of hardware, software hosting, and institutional administrative costs, but as these are both difficult to quantify and apply to all approaches they will not be included in the estimates below. The costs of each approach is summarised in Table 7.1.

[1] http://www.payscale.com/research/UK/Job=Research_Scientist/Salary

Table 7.1: Comparison of estimated costs (in US$) using four different annotation methods.

Approach	Cost (US$)/markable
Traditional, High Quality	3.00
Traditional, Medium Quality	1.20
Microworking	1.20
Games-with-a-purpose	0.47

For **Traditional, High Quality (THQ)** annotation, a formal coding scheme is developed, and often extensive agreement studies are carried out; then every document is doubly annotated according to the coding scheme by two professional annotators under the supervision of an expert, typically a linguist, and annotation is followed by merging of the annotations. It is this type of annotation which requires in the order of $1 million per one million tokens, i.e. $1 per token. Texts may typically contain around one markable every three tokens, so we get a cost of $3 per markable.

Traditional, Medium Quality (TMQ) annotation is typically carried out by trained, but not professional annotators, generally students, under the supervision of an expert annotator. Estimates for this type of work are in the order of $400,000 per one million tokens, including expert annotator costs, i.e. around $0.4 per token, or $1.2 per markable.

Costs of **microworking** depend on the amount paid per HIT and on the extent of duplication and redundancy. $0.05 per HIT is the minimum required for non-trivial tasks, and for a task such as anaphora, the cost is more like $0.1 per markable. As many as ten HITs per task are required to produce a reasonable-quality answer which results in a cost of $1 per markable, i.e. around $330,000 per million tokens. In addition, an administrator is typically required to set up the task and follow it up. This would give a total cost in the region of $380,000 per million tokens / $1.2 per markable, which is the same cost as with TMQ.

The cost per annotation for *Phrase Detectives* has been estimated to be $0.47 per markable based on a projected cost for annotating one million words [Poesio et al., 2013].[1] The cost of groupsourcing will be primarily in the data mining and processing

[1]Long term data collection efforts have a large initial upfront cost, but continue collecting data with minimal administrative oversight or expenditure until the goal is reached.

side; however, it is not possible to project an estimate as no full scale system has been built yet.

From these estimates it is clear that creating resources using traditional methods is expensive and this approach is best suited when the quality of the data are paramount.

Microworking for simple tasks is quick to set up and cheap; however, more complex tasks are more expensive. The quality of such resources needs more investigation and the approach becomes prohibitively expensive when scaling up to large resources. Microworking approaches are therefore most suited for small-to-medium scale resources, or prototyping interfaces.

The gaming approach is expensive compared to microworking to set up, but the data collection is cheap. In a long-term project it is conceivable to create large resources, with the main problem being the length of time it would take to collect the data. Over a long period of time the data collection would not only need continuous effort for player recruitment, but also the project requirements may change, requiring further development of the platform. With this in mind, this approach is most suited to a long-term, persistent data collection effort that aims to collect very large amounts of data.

Increasing efficiency and reducing costs One of the simplest ways of reducing costs is to increase the efficiency of the human computation. By optimising the data collection model this research has shown that it is possible to maintain high-quality results whilst drastically reducing the amount of human effort required. By comparing the work of annotators against annotators with an additional validation stage we showed that the latter can increase the overall quality of a crowd system without introducing more noise. This was formalised in the AV Model and demonstrated in *Phrase Detectives* and on social networks. A non-optimised model shows high quality at near-expert annotator performance; however, the cost, noise and speed are high making this method too expensive via microworking, too noisy for extracting data in high-spam scenarios and too slow for short-term data collection projects.

The AV Model can be optimised to maintain quality whilst reducing noise and increasing efficiency. The investigation showed that using agreement validation (instead of full validation or disagreement validation) increases efficiency without reducing quality. Additionally, an optimised model reduces the number of annotations that are required,

in a way so as not to affect quality significantly but also to reduce noise and cost. This reinforces the idea that understanding how many opinions need to be gathered before stopping is key to making a crowd-based system efficient.

Pre-annotation of the data and bootstrapping can reduce the task load, increase the annotation speed and quality [Fort and Sagot, 2010] and allow participants to work on more interesting tasks that are ambiguous or difficult. Bootstrapping has the downside of influencing the quality of usable output data and errors that exist in the input data multiply when used in crowdsourcing.

This was seen in *Phrase Detectives* when occasional errors in the pre-processing of a document led to some markables having an incorrect character span. The game allowed players to flag markables with errors for correction by administrators (and to skip the markable if appropriate); however, this created a bottleneck.

As can be seen from the cost breakdown of the gaming approach, more savings can be made by reusing an existing GWAP platform (the development of the Facebook version of the game cost half that of the original game) or by making a platform for multiple games (such as *Wordrobe* [Venhuizen et al., 2013]).

The advantage of a gaming approach over microworking is that personal and social incentives can be used, as well as financial, to minimise the cost and maximise the persistence of the system. The use of prizes can motivate players to contribute more whilst still offering value for money as part of a controlled budget. Conversely, a microworking approach can be much faster than other approaches because the motivational elements are more controllable, if expensive.

However, the race towards reducing costs might have a worrying side-effect as short-term microworking costs could become the standard. Funding agencies will expect low costs in future proposals and it will become hard to justify funding to produce resources with more traditional, or even GWAP-based methodologies. Another issue raised by microworking is the legal status of intellectual property rights of the resources created and some US universities have insisted on institutional review board approval for microworking experiments [Chamberlain et al., 2013].

7.9 Harnessing collective intelligence on social networks

Harnessing the collective intelligence of communities on social networks is not straightforward, but the rewards are high. If a suitable community can be found to align with the task of the requester and the data can be extracted from the network, it has shown to be a useful type of crowdsourcing approach. Aggregating the social network data in a similar way to crowdsourcing, for example using the AV Model, will allow the automatic extraction of knowledge and sophisticated crowd aggregation techniques [Raykar et al., 2010] can be used to gauge the confidence of data extracted from threads on a large scale.

A validation model is intuitive to users and features in some form on most social network platforms. Typically a 'like' or 'upvote' button can be found on messages and replies, allowing the community to show favour for particular solutions, and this method has been shown to be effective and efficient in the experimental work here. Other forms of voting exist, such as full validation (like and dislike) or graded voting (using a five star vote system) allowing for more fine-grained analysis of the community's preference; however, further research is needed to assess whether this is actually a waste of human effort and a simple like button proves to be the most effective.

In this research, users are rewarded for agreement and not punished for being disagreed with; however, other scoring models of this kind do exist [Rafelsberger and Scharl, 2009]. The social network Facebook has resisted repeated calls from users to add a dislike button for presumably this reason, especially as their content is linked to advertising. It may be that negative scoring would produce better results when using the model in post-processing or if the user did not know they were being punished. Social networks discourage the expression of negative views of other users' posts and it seems intuitive that positive behaviour be reinforced in crowdsourcing to encourage participation. The low frequency of negative statements found in the test set also suggests that correcting a user's opinion is a socially uncomfortable thing to do, even if it would improve the quality of the solution.

This research has focused on the social network Facebook because it contained examples of a defined task performed by defined groups in the network. Other systems are

of interest in community problem solving, in particular StackOverflow[1] and Github[2], or community collaboration in building large-scale accessible resources such as LinkedIn[3] or Flickr.[4] The methodology and corresponding issues discussed for groupsourcing on Facebook apply to some degree to these other types of social networks.

7.10 Limitations of a groupsourcing approach

Despite the many benefits of social networks, there are also some significant limitations.

The constantly changing underlying technology of the network, as well popularity with users, means that long-term groupsourcing projects need to spend more time adjusting their platforms to maintain compliance. Although fairly mature with a high take-up rate, social networks are still an emerging technology, and changes are made to the terms of service, access and software language that could swiftly render a dependent platform redundant.

Another drawback to using social networks is that people use them in different ways and there is no right way. There are also a proportion of user accounts used for spreading advertising or for spamming, although this is common in all crowdsourcing. Users have different expectations that may lead to segregation in groups and data not being entered in a fashion that is expected. Users can also change a post after it has received replies, meaning a user can make a task request and then change the message once a solution has been offered, even deleting replies from the thread dialogue. This is not malicious or ungrateful behaviour, but simply a different way of using groups to organise data. Users who post requests for solutions to tasks may get better answers if they create a well-formed question and provide as much metadata as possible, as the lack of both is often a cause of frustration in some social network groups.

It is unclear in the long term how social networking will continue as a popular pastime, and maintaining a community's interest in a project over time will need to be carefully managed. There may also be a saturation point of how many projects can be implemented to existing communities and this is also a problem for other peer production approaches.

[1] `http://stackoverflow.com`
[2] `https://github.com`
[3] `https://www.linkedin.com`
[4] `https://www.flickr.com`

The method of data caching described here only creates a snapshot of a group. Further development would be required to incorporate the temporal dynamics of social networks and filtering of messages would be required to minimise the database load [Maynard, Bontcheva, and Rout, 2012].

A significant challenge for groupsourcing as a methodology is the automatic processing of the threads. There are a large quantity of data associated with threads and removing this overhead is essential when processing on a large scale. The natural language processing needs to cope with ill-formed grammar and spelling, and sentences for which only context could make sense of the meaning, for example (taken from the subcorpus):

> 'And my current puzzle ...'
> 'Need assistance with this tunicate please.'
> 'couldn't find an ID based on these colours'
> 'Sven Kahlbrock please talk Latin to me ;-)'

Additionally, how successful will the automatic processing of sentiment be on such poorly formed text? Negative and compound assertions will cause problems for automatic processing; however, incidents of these in the corpora studied here were very low.

The image classification task that was investigated here uses natural language to solve the task; however, machine learning could use the image itself to classify the content. Much like the language of social networks, images also vary in quality and there is little control over what is posted. Poor-quality images or images with low illumination, unusual poses, clutter, occlusion, different viewpoints and low resolution will all make the image processing much more difficult.

This investigation of groupsourcing shows it to be a potentially useful way to complete tasks and perform data collection, but can this method be applied to other tasks? There have been examples of other tasks being completed on different social networks such as expert finding, job hunting, computer software bug fixing, etc., and these, like the image classification task examined here, are complex human computation tasks that are performed with the collective intelligence of a group. This is unlike the approach of crowdsourcing generally in which complex tasks are broken down into smaller chunks that can easily be completed by non-experts. It takes a degree of creativity to imagine

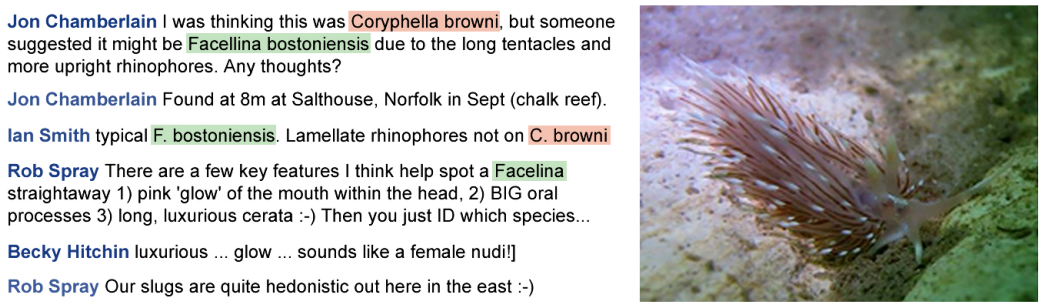


Figure 7.2: Detail of a typical message containing an image classification task having been analysed for named entities.

mundane tasks in a format that might be applicable to groupsourcing and this may be its biggest limitation.

7.11 Applications for groupsourcing

One application of this research is in response to the motivating scenario outlined in the introduction (see Section 1.1): to assess the scale and speed of coral reef degradation caused by factors such as pollution, overfishing and climate change by harnessing the collective intelligence on social networks.

Marine ecosystems are complex networks of interactions between communities of species [Paine, 1966; Sala and Sugihara, 2005]. By modelling these networks it is possible to predict how vulnerable they are to changes, such as the loss of keystone species. The degree to which a species can adapt its interactions within a community, termed *plasticity*, greatly increases its chance of survival, and the survival of the entire system, during periods of change. However, our understanding of species interactions in the traditional literature is based on limited observations. A better estimation of plasticity could be achieved by processing more sources of information.

Since the recent popularity of SCUBA diving as a recreational activity, combined with cheaper and easier to use underwater cameras, there are now a huge amount of unstructured data about marine ecosystems on the Internet. The first task is to identify the marine species (an image classification task) and then to understand the text associated with it (a text analysis task).

Ecological questions can be answered, to some extent, by looking at the range

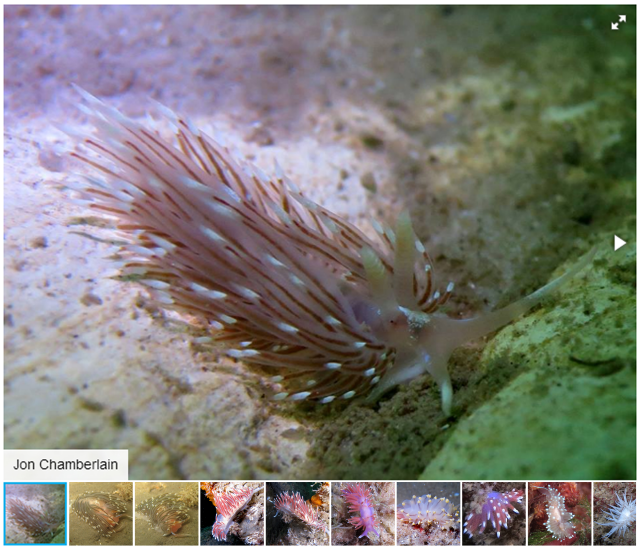


Figure 7.3: Screenshot of the *Purple Octopus* aggregated image gallery.

of conditions in which a species can exist and the interactions it has within different communities by mining social network data and resolving image classification by crowdsourcing methods.

The *Purple Octopus* prototype website The data derived from the experiments in groupsourcing have been made available to the public through a prototype website called *Purple Octopus*.[1]

In order to explore the ecological data, all text elements of the threads (messages and replies) were parsed for text strings representing marine species entities using the World Register of Marine Species (WoRMS) taxonomy[2] (see Figure 7.2).

In the same way a database of location names[3] was used to find locations mentioned within the text. There were problems caused by the structure of the ontology, the informal reporting of locations in the thread text and disambiguation with other entities and it is also the case that marine species are not found (usually) in terrestrial locations and more usual for a location to be referenced by a locally-known dive site name.

However, using this simple pattern matching, the prototype website can visualise the images and thread data of social networks with marine species and locations represented

[1] `http://www.purpleoctopus.org`
[2] `http://www.marinespecies.org`, accessed September 2012.
[3] `http://www.dbis.informatik.uni-goettingen.de/Mondial/#SQL`

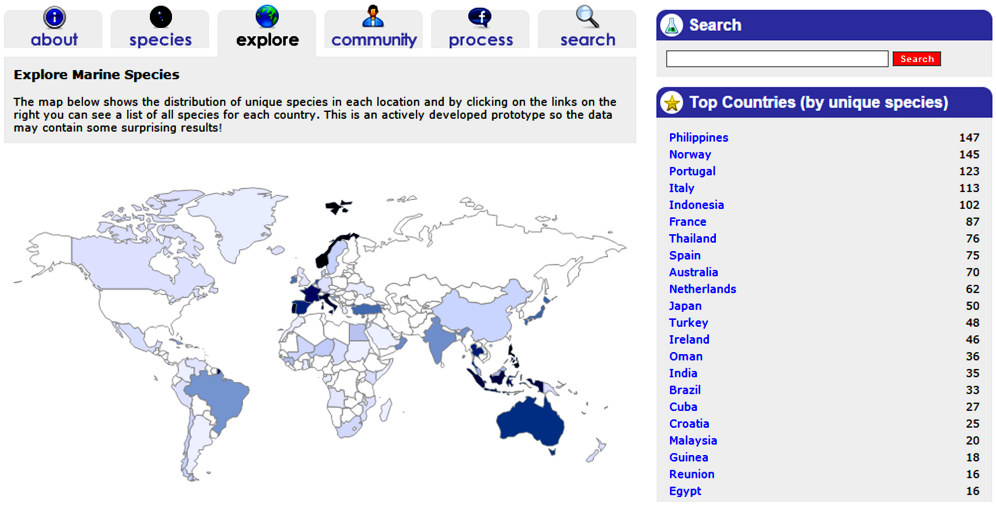


Figure 7.4: Screenshot of species richness across the groupsourced dataset.

in several ways:

- On the entity page, all messages related to a species are listed along with a gallery of photographic examples of the species (see Figure 7.3).[1]
- The entity page also shows associated species, i.e. other species named in the same threads, which indicate interaction (for example, predation or symbiosis) or morphological similarity;
- On the entity page, a map of co-mentioned locations for a species, representing its geographical distribution;
- On the explore page, a map showing species richness (total number of individual species co-mentioned with a country name) with a link to view all of the species co-mentioned with a particular country (see Figure 7.4);
- Groups in which the data were extracted and top contributors from each group, ordered by the number of posts made.

The prototype interface allowed a degree of informal testing to investigate the information extraction and to see what kind of problems that were likely to be encountered

[1]Only links to the images were stored, the images themselves are hosted on the social network. Each image was credited with the author's name.

if the groupsourcing approach were to be utilised for marine conservation in future work.

The ultimate goal is to create an accurate database of information derived from social networks that can be explored to provide actionable knowledge.

8

Conclusions

The goal of this research was to discover if collective intelligence on social networks could be used to create large-scale data resources, with high-quality labelling of information about the data, that can be used to create knowledge to solve problems that cannot currently be addressed in any other way. The research showed that social networks can be viewed as problem-solving systems, sharing common features of other crowdsourcing approaches. The benefits of using a crowd to solve problems are tempered with the many challenges this approach presents.

Social networks have large numbers of users so it is intuitive to believe that a system deployed on one would benefit from increased exposure. These issues were investigated using *Phrase Detectives*, an online game designed to collect annotations about human language, with one system deployed as a standalone game and another deployed on the social network Facebook.

Players preferred using a game deployed on a social network and do more work but, in this case, it did not translate to higher quality. It is a well-studied phenomenon that a group of non-experts can perform as well, if not better, than a single expert at problem solving and both versions of the game produce annotation decisions close to an expert opinion.

This research has also shown that a more sophisticated annotation model can be used in which the collected decisions are also validated by the users. The Annotation Validation (AV) Model is described, simulated and tested on real data from the *Phrase Detectives* game, and also data from social networks, to show that validation not only improves quality, but can also increase data collection efficiency. In particular, this

research discovered that a simple 'like' or 'upvote' decision is sufficient in an optimised model.

The research explored the idea that problem solving is an inherent part of the way humans interact with each other on social networks and that it can be viewed in the same way as a crowdsourcing system. In comparison to other methods of crowdsourcing, social networks offer a high-accuracy, data-driven and low-cost approach. Users are self-organised and intrinsically motivated to participate, with open access to the data. By archiving social network data they can be categorised and explored in meaningful ways. There are significant challenges to automatically process and aggregate data generated from social networks; however, this research shows the huge potential for this type of collective intelligence.

References

Agichtein, E.; Castillo, C.; Donato, D.; Gionis, A.; and Mishne, G. 2008. Finding high-quality content in social media. In *Proceedings of the 1st ACM International Conference on Web Search and Data Mining (WSDM'08)*, 183–194. 33, 129

Agichtein, E.; Brill, E.; and Dumais, S. 2006. Improving web search ranking by incorporating user behavior information. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'06)*, 19–26. 153

Aker, A.; El-haj, M.; Albakour, D.; and Kruschwitz, U. 2012. Assessing crowdsourcing quality through objective tasks. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC'12)*. 20, 154

Albert, P. S., and Dodd, L. E. 2008. On estimating diagnostic accuracy from studies with multiple raters and partial gold standard evaluation. *Journal of the American Statistical Association* 103(481):61–73. 30

Alon, N.; Feldman, M.; Lev, O.; and Tennenholtz, M. 2015. How robust is the wisdom of the crowds? In *Proceedings of the 24th International Joint Conference on Artificial Intelligence, IJCAI'15*, 2055–2061. 158

Alonso, O., and Mizzaro, S. 2009. Can we get rid of TREC assessors? Using Mechanical Turk for relevance assessment. In *Proceedings of 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'09) Workshop on The Future of IR Evaluation.* 23

Alonso, O.; Rose, D. E.; and Stewart, B. 2008. Crowdsourcing for relevance evaluation. *SIGIR Forum* 42(2):9–15. 23

REFERENCES

Artstein, R., and Poesio, M. 2008. Inter-coder agreement for computational linguistics. *Computational Linguistics* 34(4):555–596. 110

Attardi, G., and the Galoap Team. 2010. Phratris. Demo presented at the 9th International Semantic Web Conference (ISWC'10) tutorial INSEMTIVES'10. 38

Banko, M., and Etzioni, O. 2008. The tradeoffs between open and traditional relation extraction. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics (ACL'08)*, 28–36. 13

Banko, M.; Cafarella, M. J.; Soderland, S.; Broadhead, M.; and Etzioni, O. 2007. Open information extraction from the web. In *Proceedings of the 12th International Joint Conference on Artificial Intelligence (IJCAI'07)*, 2670–2676. 13

Bartle, R. 1996. Hearts, Clubs, Diamonds, Spades: Players who suit MUDs. *The Journal of Virtual Environments* 1(1). 40

Basile, V.; Bos, J.; Evang, K.; and Venhuizen, N. 2012. Developing a large semantically annotated corpus. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC'12)*, 3196–3200. 13

Beijbom, O.; Edmunds, P. J.; Kline, D. I.; Mitchell, B. G.; and Kriegman, D. 2012. Automated annotation of coral reef survey images. In *Proceedings of the 25th IEEE Conference on Computer Vision and Pattern Recognition (CVPR'12)*. 18

Benkler, Y., and Nissenbaum, H. 2006. Commons-based peer production and virtue. *Journal of Political Philosophy* 14(4):394–419. 31

Bernstein, M. S.; Little, G.; Miller, R. C.; Hartmann, B.; Ackerman, M. S.; Karger, D. R.; Crowell, D.; and Panovich, K. 2010. Soylent: A word processor with a crowd inside. In *Proceedings of the 23nd Annual ACM Symposium on User Interface Software and Technology (UIST'10)*, 313–322. 27

Bernstein, M. S.; Karger, D. R.; Miller, R. C.; and Brandt, J. 2012. Analytic methods for optimizing realtime crowdsourcing. *Computing Research Repository (CoRR)* abs/1204.2995. 151

Bhardwaj, V.; Passonneau, R.; Salleb-Aouissi, A.; and Ide, N. 2010. Anveshan: A tool for analysis of multiple annotators' labeling behavior. In *Proceedings of the 4th Linguistic Annotation Workshop (LAW IV)*. 33

Biederman, I. 1987. Recognition-by-components: A theory of human image understanding. *Psychological Review* 94:115–147. 16

Bigham, J. P.; Jayant, C.; Ji, H.; Little, G.; Miller, A.; Miller, R. C.; Miller, R.; Tatarowicz, A.; White, B.; White, S.; and Yeh, T. 2010. VizWiz: Nearly real-time answers to visual questions. In *Proceedings of the 23rd Annual ACM symposium on User Interface Software and Technology (UIST'10)*. 151

Bos, J., and Nissim, M. 2015. Uncovering noun-noun compound relations by gamification. In *Proceedings of the 20th Nordic Conference of Computational Linguistics (NODALIDA'15)*, 251–255. 38

Bosu, A.; Corley, C. S.; Heaton, D.; Chatterji, D.; Carver, J. C.; and Kraft, N. A. 2013. Building reputation in StackOverflow: An empirical investigation. In *Proceedings of the 10th Working Conference on Mining Software Repositories (MSR'13)*, 89–92. 29

Brabham, D. C. 2012a. Motivations for participation in a crowdsourcing application to improve public engagement in transit planning. *Journal of Applied Communication Research* 40(3):307–328. 22

Brabham, D. C. 2012b. The myth of amateur crowds. *Information, Communication and Society* 15(3):394–410. 24, 158

Brabham, D. C. 2013. *Crowdsourcing*. The MIT Press. 19, 44

Bruckman, A. 1999. Can educational be fun? In *Proceedings of the 13th Annual Game Developers Conference (GDC'99)*. 49

Bullimore, R. D.; Foster, N. L.; and Howell, K. L. 2013. Coral-characterized benthic assemblages of the deep Northeast Atlantic: Defining coral gardens to support future habitat mapping efforts. *ICES Journal of Marine Science: Journal du Conseil*. 17

Bunt, H.; Alexandersson, J.; Choe, J.-W.; Fang, A. C.; Hasida, K.; Petukhova, V.; Popescu-Belis, A.; and Traum, D. 2012. ISO 24617-2: A semantically-based standard for dialogue annotation. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC'12)*. 33, 129

Burchardt, A.; Erk, K.; Frank, A.; Kowalski, A.; Pado, S.; and Pinkal, M. 2009. FrameNet for the semantic analysis of German: Annotation, representation and automation. In Boas, H. C., ed., *Multilingual FrameNets in Computational Lexicography: Methods and Applications*. Mouton De Gruyter. 12

Burke, L.; Kura, Y.; Kassem, K.; Revenga, C.; Spalding, M.; and McAllister, D. 2001. Pilot analysis of global ecosystems: Coastal ecosystems. Technical report, World Resources Institute, Washington, DC. 1

Burnard, L. 2000. The British National Corpus reference guide. Technical report, Oxford University Computing Services, Oxford. 12

Callison-Burch, C., and Dredze, M. 2010. Creating speech and language data with Amazon's Mechanical Turk. In *Proceedings of the 11th Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL HLT 2010) Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk (CSLDAMT '10)*. 156

Callison-Burch, C. 2009. Fast, cheap, and creative: Evaluating translation quality using Amazon's Mechanical Turk. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing (EMNLP'09)*. 33

Card, S. K.; Newell, A.; and Moran, T. P. 1983. *The Psychology of Human-Computer Interaction*. L. Erlbaum Associates Inc. 153

Cesar, H., L. B., and Pet-Soede, L. 2003. The economics of worldwide coral reef degradation. Technical report, Cesar Environmental Economics Consulting (CEEC), 6828GH Arnhem, The Netherlands. 1

Chamberlain, J., and O'Reilly, C. 2014. User performance indicators in task-based data collection systems. In *Proceedings of the 2014 iConference workshop MindTheGap'14*. 8, 11, 67, 98, 145, 153

Chamberlain, J.; Fort, K.; Kruschwitz, U.; Mathieu, L.; and Poesio, M. 2013. Using games to create language resources: Successes and limitations of the approach. In *ACM Transactions on Interactive Intelligent Systems*, volume The People's Web Meets NLP: Collaboratively Constructed Language Resources. Springer. 5, 8, 11, 145, 162

Chamberlain, J.; Kruschwitz, U.; and Poesio, M. 2009. Constructing an anaphorically annotated corpus with non-experts: Assessing the quality of collaborative annotations. In *Proceedings of the 4th International Joint Conference on Natural Language Processing (IJCNLP'09) Workshop on The People's Web Meets NLP: Collaboratively Constructed Semantic Resources*. 8, 108

Chamberlain, J.; Kruschwitz, U.; and Poesio, M. 2012. Motivations for participation in socially networked collective intelligence systems. In *Proceedings of the 1st Annual Conference on Collective Intelligence (CI'12)*. 1, 8, 11, 43, 67, 150

Chamberlain, J.; Kruschwitz, U.; and Poesio, M. 2013. Methods for engaging and evaluating users of human computation systems. In *Handbook of Human Computation*. Springer. 8, 11

Chamberlain, J.; Poesio, M.; and Kruschwitz, U. 2009. A new life for a dead parrot: Incentive structures in the Phrase Detectives game. In *Proceedings of the 18th International World Wide Web Conference (WWW'09) Workshop on Web Incentives (WEBCENTIVES'09)*. 8, 20

Chamberlain, J.; Poesio, M.; and Kruschwitz, U. 2008. Phrase Detectives: A web-based collaborative annotation game. In *Proceedings of the 2008 International Conference on Semantic Systems (I-Semantics'08)*. 8, 80

Chamberlain, J. 2014a. The Annotation-Validation (AV) model: Rewarding contribution using retrospective agreement. In *Proceedings of the 1st International Workshop on Gamification for Information Retrieval (GamifIR'14)*. 8, 11, 67, 97, 145

Chamberlain, J. 2014b. Groupsourcing: Distributed problem solving using social networks. In *Proceedings of 2nd AAAI Conference on Human Computation and Crowdsourcing (HCOMP'14)*. 8, 11, 43, 62, 125, 145, 150

Chamberlain, J. 2014c. Groupsourcing: Problem solving, social learning and knowledge discovery on social networks. In *Proceedings of 2nd AAAI Conference on Human Computation and Crowdsourcing (HCOMP'14)*. 8, 11, 43, 125, 145

Chandler, D., and Kapelner, A. 2013. Breaking monotony with meaning: Motivation in crowdsourcing markets. *Journal of Economic Behavior and Organization* 90:123–133. 21

Chittka, L.; Skorupski, P.; and Raine, N. E. 2009. Speed–accuracy tradeoffs in animal decision making. *Trends in Ecology & Evolution* 24(7):400–407. 26

Chklovski, T., and Gil, Y. 2005. Improving the design of intelligent acquisition interfaces for collecting world knowledge from web contributors. In *Proceedings of the 3rd International Conference on Knowledge Capture (K-CAP'05)*. 25, 37, 73

Chklovski, T. 2005. Collecting paraphrase corpora from volunteer contributors. In *Proceedings of the 3rd International Conference on Knowledge Capture (K-CAP'05)*. 37

Clery, D. 2011. Galaxy evolution. Galaxy Zoo volunteers share pain and glory of research. *Science* 333(6039):173–5. 21, 32

Csikszentmihalyi, M. 1990. *Flow : The Psychology of Optimal Experience*. Harper and Row. 35

Csomai, A., and Mihalcea, R. 2008. Linking documents to encyclopedic knowledge. *IEEE Intelligent Systems*. Special issue on Natural Language Processing for the Web. 31

Culotta, A.; McCallum, A.; and Betz, J. 2006. Integrating probabilistic extraction models and data mining to discover relations and patterns in text. In *Proceedings of the 2006 Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics (HLT-NAACL'06)*, 296–303. 13

Culverhouse, P.; Williams, R.; Reguera, B.; Herry, V.; and González-Gil, S. 2003. Do experts make mistakes? A comparison of human and machine identification of dinoflagellates. *Marine Ecology Progress Series* 247:17–25. 18, 158

Dabbish, L.; Stuart, H. C.; Tsay, J.; and Herbsleb, J. D. 2014. Transparency and coordination in peer production. *Computing Research Repository (CoRR)* abs/1407.0377. 31

Dandapat, S.; Biswas, P.; Choudhury, M.; and Bali, K. 2009. Complex linguistic annotation - No easy way out! A case from Bangla and Hindi POS labeling tasks. In *Proceedings of the 3rd ACL Linguistic Annotation Workshop (LAW III)*. 24

Das, R., and Vukovic, M. 2011. Emerging theories and models of human computation systems: A brief survey. In *Proceedings of the 2nd International Workshop on Ubiquitous Crowdsouring (UbiCrowd'11)*, 1–4. 30, 44, 48, 50

Dawid, P., and Skene, A. M. 1979. Maximum likelihood estimation of observer error-rates using the EM algorithm. *Applied Statistics* 20–28. 29

Debelius, H., and Peyer, B. 2004. *Nudibranchs and Sea Snails: Indo-Pacific Field Guide*. IKAN-Unterwasserarchiv. 222

Deneme. 2009. How many turkers are there? `http://groups.csail.mit.edu/uid/deneme/?p=502`. 149

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. ImageNet: A large-scale hierarchical image database. In *Proceedings of the 2009 Conference on Computer Vision and Pattern Recognition (CVPR'09)*. 16

Deterding, S.; Dixon, D.; Khaled, R.; and Nacke, L. 2011. From game design elements to gamefulness: Defining "gamification". In *Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments (MindTrek'11)*, 9–15. 36

Donahue, J., and Grauman, K. 2011. Annotator rationales for visual recognition. In *Proceedings of the 13th International Conference on Computer Vision (ICCV'11)*, 1395–1402. 16

Eickhoff, C. 2014. Crowd-powered experts: Helping surgeons interpret breast cancer images. In *Proceedings of the 1st International Workshop on Gamification for Information Retrieval (GamifIR'14)*. 159

Ertekin, S.; Rudin, C.; and Hirsh, H. 2014. Approximating the crowd. *Data Mining and Knowledge Discovery* 28(5-6):1189–1221. 29

Estellés-Arolas, E., and González-Ladrón-De-Guevara, F. 2012. Towards an integrated crowdsourcing definition. *Journal of Information Science* 38(2):189–200. 19, 63

Everingham, M.; Van Gool, L.; Williams, C.; Winn, J.; and Zisserman, A. 2010. The Pascal Visual Object Classes (VOC) challenge. *International Journal of Computer Vision* 88(2):303–338. 16

Faridani, S.; Bitton, E.; Ryokai, K.; and Goldberg, K. 2010. OpinionSpace: A scalable tool for browsing online comments. In *Proceedings of the 28th ACM Conference on Human Factors in Computing Systems (CHI'10)*. 62

Fei-Fei, L.; Fergus, R.; and Perona, P. 2004. Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories. In *Proceedings of the 2004 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'04) Workshop of Generative Model Based Vision (WGMBV'04)*. 16

Fellbaum, C. 1998. *WordNet: An Electronic Lexical Database*. MIT Press. 16

Feng, D.; Besana, S.; and Zajac, R. 2009. Acquiring high quality non-expert knowledge from on-demand workforce. In *Proceedings of the 4th International Joint Conference on Natural Language Processing (IJCNLP'09) Workshop on The People's Web Meets NLP: Collaboratively Constructed Semantic Resources*. 23

Fenouillet, F.; Kaplan, J.; and Yennek, N. 2009. Serious games et motivation. In *4eme Conference francophone sur les Environnements Informatiques pour l'Apprentissage Humain (EIAH'09), vol. Actes de lAtelier "Jeux Serieux: conception et usages"*. 20

Forsyth, D. 2005. *Group Dynamics*. International student edition. Cengage Learning. 62

Fort, K.; Adda, G.; and Cohen, K. B. 2011. Amazon Mechanical Turk: Gold mine or coal mine? *Computational Linguistics (editorial)* 37:413–420. 33

Fort, K., and Sagot, B. 2010. Influence of pre-annotation on POS-tagged corpus development. In *Proceedings of the 4th ACL Linguistic Annotation Workshop (LAW IV)*. 162

Gao, H.; Wang, X.; Barbier, G.; and Liu, H. 2011. Promoting coordination for disaster relief – From crowdsourcing to coordination. In *Social Computing, Behavioral-Cultural Modeling and Prediction*, volume 6589 of *Lecture Notes in Computer Science*. Springer. 197–204. 62

Geiger, D.; Rosemann, M.; and Fielt, E. 2011. Crowdsourcing information systems: A systems theory perspective. In *Proceedings of the Australasian Conference on Information Systems (ACIS'11)*. 20, 44, 51

Gillick, D., and Liu, Y. 2010. Non-expert evaluation of summarization systems is risky. In *Proceedings of the 11th Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL HLT 2010) Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk (CSLDAMT '10)*. 156

Glott, R.; Schmidt, P.; and Ghosh, R. 2010. Wikipedia survey – Overview of results. *UNU-MERIT* 1–11. 150

Gonella, P.; Rivadavia, F.; and Fleischmann, A. 2015. Drosera magnifica (Droseraceae): the largest New World sundew, discovered on Facebook. *Phytotaxa* 220(3):257–267. 5, 41

Green, N.; Breimyer, P.; Kumar, V.; and Samatova, N. F. 2010. Packplay: Mining semantic data in collaborative games. In *Proceedings of the 4th Linguistic Annotation Workshop (LAW IV)*. 38

He, J.; van Ossenbruggen, J.; and de Vries, A. P. 2013. Do you need experts in the crowd?: A case study in image annotation for marine biology. In *Proceedings of the 10th Open Research Areas in Information Retrieval (OAIR'13)*, 57–60. 32

Heekeren, H. R.; Marrett, S.; and Ungerleider, L. G. 2008. The neural systems that mediate human perceptual decision making. *Nature reviews. Neuroscience* 9(6):467–479. 25

Henry, L., and Roberts, J. M. 2014. Recommendations for best practice in deep-sea habitat classification: Bullimore et al. as a case study. *ICES Journal of Marine Science: Journal du Conseil* 71(4):895–898. 17, 18, 158

Herdagdelen, A., and Baroni, M. 2012. Bootstrapping a game with a purpose for commonsense collection. *ACM Transactions on Intelligent Systems and Technology (TIST)* 3(4):59. 40, 148

Hitzeman, J., and Poesio, M. 1998. Long-distance pronominalisation and global focus. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics (COLING-ACL'98)*, 550–556. 72

Hladká, B.; Mírovskỳ, J.; and Schlesinger, P. 2009. Play the language: Play coreference. In *Proceedings of the 4th International Joint Conference on Natural Language Processing (IJCNLP'09)*. 38

Hobbs, J. R. 1978. Resolving pronoun references. *Lingua* 44:311–338. 72

Hoegh-Guldberg, O.; Mumby, P.; Hooten, A.; Steneck, R.; Greenfield, P.; Gomez, E.; Harvell, C.; Sale, P.; Edwards, A.; and Caldeira, K. 2007. Coral reefs under rapid climate change and ocean acidification. *Science* 318:1737. 1

Holt, B. G.; Rioja-Nieto, R.; MacNeil, A. M.; Lupton, J.; and Rahbek, C. 2013. Comparing diversity data collected using a protocol designed for volunteers with results from a professional alternative. *Methods in Ecology and Evolution* 4(4):383–392. 32

Hong, J., and Baker, C. F. 2011. How good is the crowd at "real" WSD? In *Proceedings of the 5th Linguistic Annotation Workshop (LAW V)*. 154

Hovy, E.; Marcus, M.; Palmer, M.; Ramshaw, L.; and Weischedel, R. 2006. Ontonotes: The 90% solution. In *Proceedings of the 2006 Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics (HLT-NAACL'06)*, 57–60. 12

Howe, J. 2008. *Crowdsourcing: Why the power of the crowd is driving the future of business*. Crown Publishing Group. 4, 18, 62

Hung, N. Q. V.; Tam, N. T.; Tran, L. N.; and Aberer, K. 2013. An evaluation of aggregation techniques in crowdsourcing. In Lin, X.; Manolopoulos, Y.; Srivastava, D.; and Huang, G., eds., *Web Information Systems Engineering (WISE 2013)*, volume 8181 of *Lecture Notes in Computer Science*. Springer. 1–15. 28

Hutchinson, G. E. 1957. Concluding remarks. *Cold Spring Harbor Symposia on Quantitative Biology* 22:415–427. 146

Ipeirotis, P. G.; Provost, F.; and Wang, J. 2010. Quality management on Amazon Mechanical Turk. In *Proceedings of the ACM SIGKDD Workshop on Human Computation (HCOMP'10)*, 64–67. 29

Ipeirotis, P. 2010a. Analyzing the Amazon Mechanical Turk marketplace. CeDER Working Papers. 148

Ipeirotis, P. 2010b. Demographics of Mechanical Turk. CeDER Working Papers. 148, 150

Johnson, R. F., and Gosliner, T. M. 2012. Traditional taxonomic groupings mask evolutionary history: A molecular phylogeny and new classification of the chromodorid nudibranchs. *PLoS ONE* 7(4). 147

Johnson, N. L.; Rasmussen, S.; Joslyn, C.; Rocha, L.; Smith, S.; and Kantor, M. 1998. Symbiotic Intelligence: Self-organizing knowledge on distributed networks driven by human interaction. In *Proceedings of the 6th International Conference on Artificial Life*. 4

Jovian, L. T., and Amprimo, O. 2011. OCR correction via human computational game. In *Proceedings of the 44th Hawaii International Conference on System Sciences (HICSS'11)*, 1–10. 40

Jurafsky, D., and Martin, J. H. 2008. *Speech and Language Processing (2^{nd} edition)*. Prentice-Hall. 24

Jurgens, D., and Navigli, R. 2014. It's all fun and games until someone annotates: Video games with a purpose for linguistic annotation. *Transactions of the Association for Computational Linguistics* 2:449–464. 37

Kabadjov, M. A. 2007. *Task-oriented evaluation of anaphora resolution.* Ph.D. Dissertation, University of Essex. 203

Kamar, E.; Hacker, S.; and Horvitz, E. 2012. Combining human and machine intelligence in large-scale crowdsourcing. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems (AAMAS'12)*, 467–474. 29

Kamp, H., and Reyle, U. 1993. *From Discourse to Logic.* Reidel. 14

Kanefsky, B.; Barlow, N.; and Gulick, V. 2001. Can distributed volunteers accomplish massive data analysis tasks? *Lunar and Planetary Science* XXXII. 23

Karger, D. R.; Oh, S.; and Shah, D. 2011. Iterative learning for reliable crowdsourcing systems. In Shawe-Taylor, J.; Zemel, R.; Bartlett, P.; Pereira, F.; and Weinberger, K., eds., *Advances in Neural Information Processing Systems 24*. Curran Associates, Inc. 1953–1961. 30

Kaufmann, N.; Schulze, T.; and Veit, D. 2011. More than fun and money. Worker motivation in crowdsourcing - A study on Mechanical Turk. In *Proceedings of the 17th Americas Conference on Information Systems (AMCIS'11).* 22

Kay, L. M.; Beshel, J.; and Martin, C. 2006. When good enough is best. *Neuron* 51(3):277–278. 26

Kazai, G.; Milic-Frayling, N.; and Costello, J. 2009. Towards methods for the collective gathering and quality control of relevance assessments. In *Proceedings of the 32nd International ACM Conference on Research and Development in Information Retrieval (SIGIR'09).* 23

Khattak, F. K., and Salleb-aouissi, A. 2011. Quality control of crowd labeling through expert evaluation. In *Proceedings Second Workshop on Computational Social Science and the Wisdom of Crowds (NIPS'11).* 29

Kincaid, J. P.; Fishburne, R. P.; Rogers, R. L.; and Chissom, B. S. 1975. Derivation of new readability formulas (Automated Readability Index, Fog Count and Flesch Reading Ease Formula) for Navy enlisted personnel. Technical report. 215

Klein, D.; Smarr, J.; Nguyen, H.; and Manning, C. D. 2003. Named entity recognition with character-level models. In *Proceedings of the 7th Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics (HLT-NAACL'03)*, 180–183. 13

Koller, A.; Striegnitz, K.; Gargett, A.; Byron, D.; Cassell, J.; Dale, R.; Moore, J.; and J.Oberlander. 2010. Report on the 2nd NLG challenge on generating instructions in virtual environments (GIVE-2). In *Proceedings of the 6th International Conference on Natural Language Generation (INLG'10)*. 38

Koster, R. 2005. *A Theory of Fun for Game Design*. Paraglyph. 35

Krötzsch, M.; Vrandečič, D.; Völkel, M.; Haller, H.; and Studer, R. 2007. Semantic Wikipedia. *Journal of Web Semantics* 5:251–261. 31

Kucera, H., and Francis, W. N. 1967. *Computational Analysis of Present-day American Engish*. Brown University Press. 12

Kuncheva, L.; Whitaker, C.; Shipp, C.; and Duin, R. 2003. Limits on the majority vote accuracy in classifier fusion. *Pattern Analysis & Applications* 6(1):22–31. 28

Kuo, Y.-l.; Lee, J.-C.; Chiang, K.-y.; Wang, R.; Shen, E.; Chan, C.-w.; and Hsu, J. Y.-j. 2009. Community-based game design: Experiments on social games for commonsense data collection. In *Proceedings of the ACM SIGKDD Workshop on Human Computation (HCOMP'09)*, 15–22. 40

Lafourcade, M. 2007. Making people play for lexical acquisition. In *Proceedings of the 7th Symposium on Natural Language Processing (SNLP'07)*. 38

Lakhani, K. R.; Jeppesen, L. B.; Lohse, P. A.; and Panetta, J. A. 2007. The value of openness in scientific problem solving. Working Paper 07-050, Harvard Business School. 31

Laniado, D.; Castillo, C.; Kaltenbrunner, A.; and Fuster-Morell, M. 2012. Emotions and dialogue in a peer-production community: The case of Wikipedia. In *Proceedings of the 8th International Symposium on Wikis and Open Collaboration (WikiSym'12)*. 150

Lee, K.; Caverlee, J.; and Webb, S. 2010. The social honeypot project: Protecting online communities from spammers. In *Proceedings of the 19th International Conference on World Wide Web (WWW'10)*, 1139–1140. 28

Levy, P. 1997. *Collective Intelligence: Mankind's Emerging World in Cyberspace (Helix Books)*. Perseus Books Group. 18

Lieberman, H.; Smith, D. A.; and Teeters, A. 2007. Common consensus: A web-based game for collecting commonsense goals. In *Proceedings of the 12th International Conference on Intelligent User Interfaces (IUI'07)*. 20

Lu, D., and Weng, Q. 2007. A survey of image classification methods and techniques for improving classification performance. *International Journal of Remote Sensing* 28(5):823–870. 15

Luo, X., and Shinaver, J. 2009. MultiRank: Reputation ranking for generic semantic social networks. In *Proceedings of the 18th International World Wide Web Conference (WWW'09) Workshop on Web Incentives (WEBCENTIVES'09)*. 25

Macdonald, C.; Tonellotto, N.; and Ounis, I. 2012. Learning to predict response times for online query scheduling. In *Proceedings of the 35th International ACM Conference on Research and Development in Information Retrieval (SIGIR'12)*, 621–630. 26

Malone, T.; Laubacher, R.; and Dellarocas, C. 2009. Harnessing crowds: Mapping the genome of collective intelligence. Research Paper No. 4732-09, Sloan School of Management, Massachusetts Institute of Technology. 18, 20, 30, 44, 45, 46, 49

Marcus, M.; Santorini, B.; and Marcinkiewicz, M. A. 1993. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics* 19(2):313–330. 13

Marge, M.; Banerjee, S.; and Rudnicky, A. I. 2010. Using the Amazon Mechanical Turk for transcription of spoken language. In *Proceedings of the 2010 IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP'10)*. 156

Mason, W., and Watts, D. J. 2009. Financial incentives and the "performance of crowds". In *Proceedings of the ACM SIGKDD Workshop on Human Computation (HCOMP'09)*. 20, 150

Maynard, D.; Bontcheva, K.; and Rout, D. 2012. Challenges in developing opinion mining tools for social media. *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC'12) Workshop @NLP can u tag #user_generated_content*. 165

McCallum, A.; Freitag, D.; and Pereira, F. C. N. 2000. Maximum entropy markov models for information extraction and segmentation. In *Proceedings of the 17th International Conference on Machine Learning (ICML'00)*, 591–598. 13

Michael, D. R., and Chen, S. L. 2005. *Serious Games: Games That Educate, Train, and Inform*. Muska & Lipman/Premier-Trade. 36

Miller, N.; Resnick, P.; and Zeckhauser, R. 2005. Eliciting informative feedback: The Peer-Prediction method. *Management Science* 51(9):1359–1373. 27

Mintz, M.; ; Bills, S.; Snow, R.; and Jurafsky, D. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing (IJCNLP'09)*, 1003–1011. 13

Mrozinski, J.; Whittaker, E.; and Furui, S. 2008. Collecting a why-question corpus for development and evaluation of an automatic QA-system. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL'08:HLT)*. 20

Navigli, R., and Ponzetto, S. P. 2012. BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence* 193:217–250. 16

Nov, O. 2007. What motivates Wikipedians? *Communications of the ACM* 50(11):60–64. 148

Novotney, S., and Callison-Burch, C. 2010. Cheap, fast and good enough: Automatic speech recognition with non-expert transcription. In *Proceedings of the 11th Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL HLT 2010)*. 33

Organisciak, Peter; Twidale, M. 2015. Design facets of crowdsourcing. In *Proceedings of the 2015 iConference*. 20, 44, 47, 48, 50, 51

Paine, R. T. 1966. Food web complexity and species diversity. *American Naturalist* 100:65–75. 166

Parameswaran, M., and Whinston, A. B. 2007. Social computing: An overview. *Communications of the Association for Information Systems* 19. 39

Pareto, V. 1896. *Cours d'Economie Politique*. Droz. 23

Passonneau, R. J., and Carpenter, B. 2013. The benefits of a model of annotation. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, 187–195. 30

Petrov, S.; Barrett, L.; Thibaux, R.; and Klein, D. 2006. Learning accurate, compact, and interpretable tree annotation. In *Proceedings of the 21st International Conference on Computational Linguistics*, 433–440. 205

Picton, B., and Morrow, C. 1994. *A field guide to the nudibranchs of the British Isles*. Immel Publishing. 222

Poesio, M., and Artstein, R. 2008. Anaphoric annotation in the ARRAU corpus. In *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC'08)*. 4, 12, 70

Poesio, M., and Vieira, R. 1998. A corpus-based investigation of definite description use. *Computational Linguistics* 24(2):183–216. 13

Poesio, M.; Sturt, P.; Arstein, R.; and Filik, R. 2006. Underspecification and anaphora: Theoretical issues and preliminary evidence. *Discourse Processes* 42(2):157–175. 15

Poesio, M.; Diewald, N.; Stührenberg, M.; Chamberlain, J.; Jettka, D.; Goecke, D.; and Kruschwitz, U. 2011. Markup infrastructure for the Anaphoric Bank: Supporting web collaboration. In Mehler, A.; Kühnberger, K.-U.; Lobin, H.; Lüngen, H.; Storrer, A.; and Witt, A., eds., *Modeling, Learning, and Processing of Text Technological Data Structures*, volume 370 of *Studies in Computational Intelligence*. Springer. 175–195. 201

Poesio, M.; Chamberlain, J.; Kruschwitz, U.; Robaldo, L.; and Ducceschi, L. 2013. Phrase Detectives: Utilizing collective intelligence for internet-scale language resource creation. *ACM Transactions on Interactive Intelligent Systems.* 1, 8, 11, 67, 107, 145, 160

Poesio, M. 2004a. Discourse annotation and semantic annotation in the GNOME corpus. In *Proceedings of the 2004 ACL Workshop on Discourse Annotation.* 12, 108

Poesio, M. 2004b. The MATE/GNOME scheme for anaphoric annotation, revisited. In *Proceedings of the 5th SIGdial Workshop on Discourse and Dialogue (SIGDIAL'04).* 204

Ponzetto, S., and Strube, M. 2007. Knowledge derived from Wikipedia for computing semantic relatedness. *Journal of Artificial Intelligence Research* 30:181–212. 31

Pradhan, S. S.; Ramshaw, L.; Weischedel, R.; MacBride, J.; and Micciulla, L. 2007. Unrestricted coreference: Identifying entities and events in OntoNotes. In *Proceedings of the 1st IEEE International Conference on Semantic Computing (ICSC'07).* 12, 70

Prestopnik, N.; Crowston, K.; and Wang, J. 2014. Exploring data quality in games with a purpose. In *Proceedings of the 2014 iConference.* 37, 157

Prince, E. F. 1992. The ZPG letter: Subjects, definiteness, and information-status. *Discourse description: Diverse analyses of a fund raising text* 295–325. 14

Quinn, A. J., and Bederson, B. B. 2011. Human computation: A survey and taxonomy of a growing field. In *Proceedings of the 2011 SIGCHI Conference on Human Factors in Computing Systems (CHI'11)*, 1403–1412. 25, 30, 44, 47, 48, 50, 52, 53

Raddick, M. J.; Bracey, G.; Gay, P. L.; Lintott, C. J.; Cardamone, C.; Murray, P.; Schawinski, K.; Szalay, A. S.; and Vandenberg, J. 2008. Galaxy Zoo: Motivations of citizen scientists. *American Astronomical Society Meeting Abstracts #212* 40:240. 53

Radlinski, F., and Craswell, N. 2010. Comparing the sensitivity of information retrieval metrics. In *Proceedings of the 33rd international ACM Conference on Research and Development in Information Retrieval (SIGIR'10)*, 667–674. 152

Rafelsberger, W., and Scharl, A. 2009. Games with a purpose for social networking platforms. In *Proceedings of the 20th ACM Conference on Hypertext and Hypermedia*. 40, 163

Raykar, V. C.; Yu, S.; Zhao, L. H.; Jerebko, A.; Florin, C.; Valadez, G. H.; Bogoni, L.; and Moy, L. 2009. Supervised learning from multiple experts: Whom to trust when everyone lies a bit. In *Proceedings of the 26th Annual International Conference on Machine Learning (ICML'09)*, 889–896. 30

Raykar, V. C.; Yu, S.; Zhao, L. H.; Valadez, G. H.; Florin, C.; Bogoni, L.; and Moy, L. 2010. Learning from crowds. *Journal of Machine Learning Research* 11:1297–1322. 30, 163

Recasens, M.; Màrquez, L.; Sapena, E.; Martí, M. A.; Taulé, M.; Hoste, V.; Poesio, M.; and Versley, Y. 2010. Semeval-2010 task 1: Coreference resolution in multiple languages. In *Proceedings of 5th International Workshop on Semantic Evaluations (SEMEVAL'10)*. 70

Rigby, S., and Ryan, R. M. 2011. *Glued to games: How video games draw us in and hold us spellbound*. Praeger. 22

Roberts, L. G. 1963. *Machine Perception of Three-Dimensional Solids*. Outstanding Dissertations in the Computer Sciences. Garland Publishing, New York. 16

Rokicki, M.; Zerr, S.; and Siersdorfer, S. 2015. Groupsourcing: Team competition designs for crowdsourcing. In *Proceedings of the 24th International Conference on World Wide Web (WWW'15)*, 906–915. 20, 23, 62

Ross, J.; Irani, L.; Silberman, M. S.; Zaldivar, A.; and Tomlinson, B. 2010. Who are the crowdworkers?: Shifting demographics in Mechanical Turk. In *Proceedings of the 2010 SIGCHI Conference on Human Factors in Computing Systems (CHI'10)*. 150

Rouse, A. C. 2010. A preliminary taxonomy of crowdsourcing. In *Proceedings of the 2010 Australian Conference on Information Systems (ACIS'10)*. 20, 44, 52

Russell, B. C.; Torralba, A.; Murphy, K. P.; and Freeman, W. T. 2008. LabelMe: A database and web-based tool for image annotation. *International Journal of Computer Vision* 77(1-3):157–173. 17

Ryan, R. M., and Deci, E. L. 2000. Intrinsic and extrinsic motivations: Classic definitions and new directions. *Contemporary Educational Psychology* 25(1):54 – 67. 20

Sala, E., and Sugihara, G. 2005. Food-web theory provides guidelines for marine conservation. In Belgrano, A.; Scharler, U. M.; Dunne, J.; and Ulanowicz, R. E., eds., *Aquatic Food Webs: An Ecosystem Approach*. Oxford University Press. chapter 13. 166

Schenk, E., and Guittard, C. 2011. Towards a characterization of crowdsourcing practices. *Journal of Innovation Economics* 7:93–107. 20, 44, 45, 47, 51

Schoening, T.; Bergmann, M.; Purser, A.; Dannheim, J.; Gutt, J.; and Nattkemper, T. W. 2012. Semi-automated image analysis for the assessment of megafaunal densities at the Arctic deep-sea observatory HAUSGARTEN. *PLoS ONE* 7(6). 18

Seaborn, K.; Pennefather, P.; and Fels, D. 2013. Reimagining leaderboards: Towards gamifying competency models through social game mechanics. In *Proceedings of Gamification 2013: Gameful Design, Research, and Applications*, 107–110. 36

Settles, B. 2009. Active learning literature survey. Computer Science Technical Report 1648, University of Wisconsin. 13

Sheng, V. S.; Provost, F.; and Ipeirotis, P. G. 2008. Get another label? Improving data quality and data mining using multiple, noisy labelers. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'08)*, 614–622. 28

Sidlauskas, B.; Bernard, C.; Bloom, D.; Bronaugh, W.; Clementson, M.; and Vari, R. P. 2011. Ichthyologists hooked on Facebook. *Science* 332(6029):537. 5, 41

Singh, P. 2002. The public acquisition of commonsense knowledge. In *Proceedings of the AAAI Spring Symposium on Acquiring (and Using) Linguistic (and World) Knowledge for Information Access*. 31

Siorpaes, K., and Hepp, M. 2008. OntoGame: Weaving the semantic web by online games. In Bechhofer, S.; Hauswirth, M.; Hoffmann, J.; and Koubarakis, M., eds.,

The Semantic Web: Research and Applications, volume 5021 of *Lecture Notes in Computer Science*. Springer. 751–766. 38

Smadja, F. 2009. Mixing financial, social and fun incentives for social voting. In *Proceedings of the 18th International World Wide Web Conference (WWW'09) Workshop on Web Incentives (WEBCENTIVES'09)*. 23

Snow, R.; O'Connor, B.; Jurafsky, D.; and Ng, A. Y. 2008. Cheap and fast - but is it good?: Evaluating non-expert annotations for natural language tasks. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing (EMNLP'08)*. 33, 53

Sternberg, S. 1969. The discovery of processing stages: Extensions of Donders' method. *Acta Psychologica* 30:276–315. 25

Stringhini, G.; Kruegel, C.; and Vigna, G. 2010. Detecting spammers on social networks. In *Proceedings of the 26th Annual Computer Security Applications Conference (ACSAC'10)*, 1–9. 6, 67

Stührenberg, M., and Goecke, D. 2008. SGF - An integrated model for multiple annotations and its application in a linguistic domain. In *Proceedings of the 2008 Balisage: The Markup Conference*. 201

Stührenberg, M.; Goecke, D.; Diewald, N.; Mehler, A.; and Cramer, I. 2007. Web-based annotation of anaphoric relations and lexical chains. In *Proceedings of the 2007 ACL Linguistic Annotation Workshop (LAW'07)*, 140–147. 201

Su, Q.; Pavlov, D.; Chow, J.-H.; and Baker, W. C. 2007. Internet-scale collection of human-reviewed data. In *Proceedings of the 16th International Conference on World Wide Web (WWW'07)*, 231–240. 33

Sun, Y., and Dance, C. R. 2012. When majority voting fails: Comparing quality assurance methods for noisy human computation environment. *Computing Research Repository (CoRR)* abs/1204.3516. 28

Surowiecki, J. 2005. *The Wisdom of Crowds*. Anchor. 4, 20, 28

Sweetser, P., and Wyeth, P. 2005. Gameflow: A model for evaluating player enjoyment in games. *Computer Entertainment* 3. 35

Thaler, S.; Siorpaes, K.; Simperl, E.; and Hofer, C. 2011. A survey on games for knowledge acquisition. Technical Report STITR2011-05-01, Semantic Technology Institute. 38

Torralba, A.; Fergus, R.; and Freeman, W. 2008. 80 million tiny images: A large data set for nonparametric object and scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30(11):1958–1970. 16

Tratz, S., and Hovy, E. 2010. A taxonomy, dataset, and classifier for automatic noun compound interpretation. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL'10)*. 154

van Mierlo, T. 2014. The 1% rule in four digital health social networks: An observational study. *Journal of Medical Internet Research* 16(2):33. 23

Vannella, D.; Jurgens, D.; Scarfini, D.; Toscani, D.; and Navigli, R. 2014. Validating and extending semantic knowledge bases using video games with a purpose. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL'14)*, 1294–1304. 37

Venhuizen, N.; Basile, V.; Evang, K.; and Bos, J. 2013. Gamification for word sense labeling. In *Proceedings of the 10th International Conference on Computational Semantics (IWCS'13)*, 397–403. 38, 162

Vieira, R., and Poesio, M. 2000. An empirically based system for processing definite descriptions. *Computational Linguistics* 26:539–593. 72

Vlachos, A. 2006. Active annotation. In *Proceedings of the 11th Conference of the European Chapter of the Association of Computational Linguistics (EACL'06) Workshop on Adaptive Text Extraction and Mining*. 13

von Ahn, L., and Dabbish, L. 2008. Designing games with a purpose. *Communications of the ACM* 51(8):58–67. 34, 78, 151

von Ahn, L.; Maurer, B.; McMillen, C.; Abraham, D.; and Blum, M. 2008. re-CAPTCHA: Human-based character recognition via web security measures. *Science* 321(5895):1465–1468. 26

von Ahn, L. 2006. Games with a purpose. *Computer* 39(6):92–94. 4, 25, 34, 37

Wais, P.; Lingamneni, S.; Cook, D.; Fennell, J.; Goldenberg, B.; Lubarov, D.; Marin, D.; and Simons, H. 2010. Towards building a high-quality workforce with Mechanical Turk. In *Proceedings of Computational Social Science and the Wisdom of Crowds (NIPS)*. 33

Wang, A.; Hoang, C. D. V.; and Kan, M. Y. 2010. Perspectives on crowdsourcing annotations for natural language processing. *Language Resources And Evaluation* 1–19. 30

Welinder, P.; Branson, S.; Mita, T.; Wah, C.; Schroff, F.; Belongie, S.; and Perona, P. 2010a. Caltech-UCSD Birds 200. Technical Report CNS-TR-2010-001, California Institute of Technology. 17

Welinder, P.; Branson, S.; Perona, P.; and Belongie, S. J. 2010b. The multidimensional wisdom of crowds. In Lafferty, J.; Williams, C.; Shawe-Taylor, J.; Zemel, R.; and Culotta, A., eds., *Advances in Neural Information Processing Systems 23*. Curran Associates, Inc. 2424–2432. 30

Whitehill, J.; Ruvolo, P.; fan Wu, T.; Bergsma, J.; and Movellan, J. 2009. Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. In Bengio, Y.; Schuurmans, D.; Lafferty, J.; Williams, C. K. I.; and Culotta, A., eds., *Advances in Neural Information Processing Systems 22*, 2035–2043. 30

Wiggins, A., and Crowston, K. 2011. From conservation to crowdsourcing: A typology of citizen science. In *Proceedings of the 44th Hawaii International Conference on System Sciences (HICSS'11)*, 1–10. 44

Wilkinson, C. 2008. Status of coral reefs of the world 2008. Technical report, Global Coral Reef Monitoring Network & Reef and Rainforest Research Centre. 1

Woolley, A. W.; Chabris, C. F.; Pentland, A.; Hashmi, N.; and Malone, T. W. 2010. Evidence for a collective intelligence factor in the performance of human groups. *Science* 330:686–688. 150

Xiao, J.; Hays, J.; Ehinger, K. A.; Oliva, A.; and Torralba, A. 2010. Sun database: Large-scale scene recognition from abbey to zoo. In *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'14)*, 3485–3492. 16

Yang, H., and Lai, C. 2010. Motivations of Wikipedia content contributors. *Computers in Human Behavior* 26. 21, 31

Yeun, C. A.; Noll, M. G.; Gibbins, N.; Meinel, C.; and Shadbolt, N. 2009. On measuring expertise in collaborative tagging systems. In *Proceedings of the 2009 Web Science Conference (WebSci'09)*. 25

Yuen, M.; Chen, L.; and King, I. 2009. A survey of human computation systems playing / having fun. *Information Sciences*. 30

Zaenen, A. 2006. Mark-up barking up the wrong tree. *Computational Linguistics* 32(4):577–580. 13

Zichermann, G., and Cunningham, C. 2011. *Gamification by Design: Implementing Game Mechanics in Web and Mobile Apps*. O'Reilly Media, Inc. 36

Zipf, G. K. 1949. *Human Behavior and the Principle of Least Effort*. Addison-Wesley. 23

Zwass, V. 2010. Co-creation: Toward a taxonomy and an integrated research perspective. *International Journal of Electronic Commerce* 15(1):11–48. 46

Appendix A

Examples of games-with-a-purpose

Table A.1: Categories of GWAPs with links where available.

GWAP name	URL
Image annotation	
ESP Game	http://www.gwap.com/gwap/gamesPreview/espgame
Matchin	http://www.gwap.com/gwap/gamesPreview/matchin
FlipIt	http://www.gwap.com/gwap/gamesPreview/flipit
Phetch	http://www.peekaboom.org/phetch
Peekaboom	http://www.peekaboom.org
Squigl	http://www.gwap.com/gwap/gamesPreview/squigl
Magic Bullet	http://homepages.cs.ncl.ac.uk/jeff.yan/mb.htm
Picture This	http://picturethis.club.live.com
Video annotation	
OntoTube	http://ontogame.sti2.at/games
PopVideo	http://www.gwap.com/gwap/gamesPreview/popvideo
Yahoo's VideoTagGame	http://sandbox.yahoo.com/VideoTagGame
Waisda	http://www.waisda.nl
Audio annotation	
Herd It	http://apps.facebook.com/herd-it
Tag a Tune	http://www.gwap.com/gwap/gamesPreview/tagatune
WhaleFM	http://whale.fm
Biomedical	
Foldit	http://fold.it/portal
Phylo	http://phylo.cs.mcgill.ca
EteRNA	http://eterna.cmu.edu
Transcription	
Ancient Lives	http://ancientlives.org
Old Weather	http://www.oldweather.org
Search results	
Page Hunt	http://pagehunt.msrlivelabs.com/PlayPageHunt.aspx
Social bookmarking	
Collabio	http://research.microsoft.com/en-us/um/redmond/groups/cue/collabio
Behavioural change	
Power House	http://powerhouse.stanford.edu

Table A.2: Categories of GWAPs used for NLP with links where available.

GWAP name	URL
Knowledge acquisition	
1001 Paraphrases	
LEARNER	
FACTory	http://game.cyc.com
Verbosity	http://www.gwap.com/gwap/gamesPreview/verbosity
Categorilla	http://www.doloreslabs.com/stanfordwordgame/categorilla.html
Free Association	http://www.doloreslabs.com/stanfordwordgame/freeAssociation.html
Text annotation	
Phrase Detectives	http://www.phrasedetectives.com
Phrase Detectives on Facebook	http://apps.facebook.com/phrasedetectives
PlayCoref	
PhraTris	http://galoap.codeplex.com
PackPlay	
Wordrobe	http://www.wordrobe.org
Sentiment analysis	
Sentiment Quiz	http://apps.facebook.com/sentiment-quiz
Generation	
GIVE games	http://www.give-challenge.org
Ontology building	
JeuxDeMots	http://www.jeuxdemots.org
AKI	http://www.jeuxdemots.org/AKI.php
OntoGame	http://ontogame.sti2.at/games

Appendix B

Player recruitment and financial incentives

Table B.1: The influence of financial prizes on player recruitment and activity in *Phrase Detectives* in the first 24 months of release.

Month	Prize fund	Total work	New players	Active players	Work per active player	Work per unit cost
Dec-08	225	26,142	59	48	544.6	116.2
Jan-09	225	51,206	171	97	527.9	227.6
Feb-09	210	48,734	107	66	738.4	232.1
Mar-09	210	57,303	154	63	909.6	272.9
Apr-09	225	87,593	159	66	1,327.2	389.3
May-09	180	103,866	57	32	3,245.8	577.0
Jun-09	150	57,767	61	41	1,409	385.1
Jul-09	150	67,320	48	33	2,040	448.8
Aug-09	150	61,371	35	36	1,704.8	409.1
Sep-09	150	31,117	37	34	915.2	207.4
Oct-09	150	23,912	30	27	885.6	159.4
Nov-09	150	56,016	49	30	1,867.2	373.4
Dec-09	300	105,577	123	80	1,319.7	351.9
Jan-10	90	209,593	480	297	705.7	2,328.8
Feb-10	780	186,640	147	111	1,681.4	239.3
Mar-10	0	98,031	122	62	1,581.1	-
Apr-10	0	36,231	79	40	905.8	-
May-10	0	20,099	163	32	628.1	-
Jun-10	200	67,876	384	91	745.9	339.4
Jul-10	100	32,244	120	33	977.1	322.4
Aug-10	0	11,369	95	17	668.8	-
Sep-10	0	10,316	141	15	687.7	-
Oct-10	100	48,743	205	35	1,392.7	487.4
Nov-10	0	9,051	238	15	603.4	-
Average (mean)						437.1 (SD 486.6)
Pearson's R		0.566	-0.034	0.257	0.248	
p-value		p<0.01				
Spearman's R		0.486	-0.044	0.613	0.176	
p-value		p<0.05		p<0.01		

Table B.2: The influence of financial prizes on player recruitment and activity in *Phrase Detectives* on Facebook in the first 24 months of release.

Month	Prize fund	Total work	New players	Active players	Work per active player	Work per unit cost
Feb-11	0	6,105	91	24	254.4	-
Mar-11	0	7,811	30	13	600.8	-
Apr-11	0	3,136	24	6	522.7	-
May-11	0	3,447	30	10	344.7	-
Jun-11	0	2,997	17	9	333.0	-
Jul-11	215	144,699	164	46	3,145.6	673.0
Aug-11	105	23,531	23	15	1,568.7	224.1
Sep-11	100	30,628	19	15	2,041.9	306.3
Oct-11	110	146,648	38	22	6,665.8	1,333.2
Nov-11	110	96,276	33	18	5,348.7	875.2
Dec-11	105	49,459	81	20	2,473.0	471.0
Jan-12	110	44,486	24	13	3,422.0	404.4
Feb-12	105	40,226	37	12	3,352.2	383.1
Mar-12	80	23,374	13	11	2,124.9	292.2
Apr-12	110	28,847	13	12	2,403.9	262.2
May-12	110	23,827	13	14	1,701.9	216.6
Jun-12	110	20,116	8	14	1,436.9	182.9
Jul-12	110	15,039	30	13	1,156.8	136.7
Aug-12	110	31,060	19	11	2,823.6	282.4
Sep-12	105	13,985	17	13	1,075.8	133.2
Oct-12	0	16,728	14	10	1,672.8	-
Nov-12	0	14,496	0	9	1,610.7	-
Dec-12	0	11,199	0	6	1,866.5	-
Jan-13	100	15,288	0	8	1,911.0	152.9
Average (mean)						395.6 (SD 320.7)
Pearson's R		0.648	0.397	0.594	0.553	
p-value		p<0.01		p<0.01	p<0.01	
Spearman's R		0.777	0.219	0.565	0.636	
p-value		p<0.01		p<0.01	p<0.01	

Appendix C

Technical details of *Phrase Detectives*

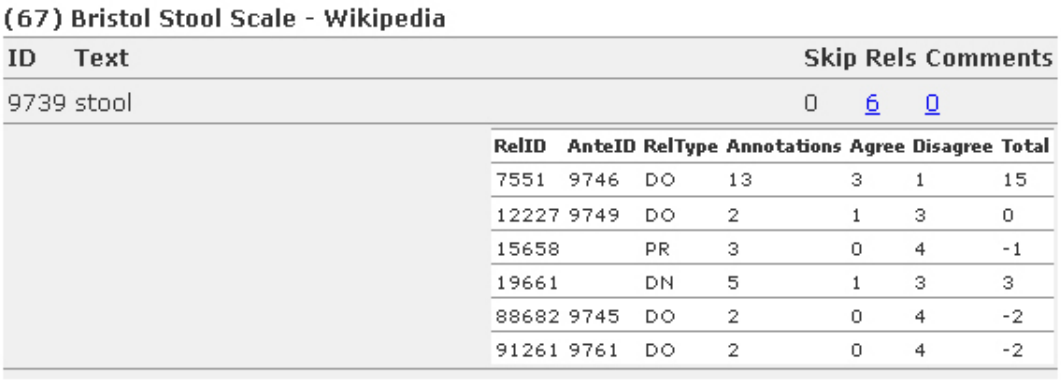
In the initial plans for the *AnaWiki* collaborative language annotation project, two types of Web collaboration would be supported: through a game-with-a-purpose for casual users eventually called *Phrase Detectives*, and through an online annotation system developed by the University of Bielefeld called *Serengeti* [Stührenberg et al., 2007]. Both types of data would be stored in a single database. As a result, the data were stored in a MySQL database the design of which is based on the *Serengeti* database, and new additions to the corpus are entered through the *Serengeti* interface [Poesio et al., 2011; Stührenberg and Goecke, 2008].

The *Phrase Detectives* game was built primarily in PHP, HTML, CSS and JavaScript. The overall design was created to conform to Internet usability, accessibility and compatibility standards. The design incorporates licensed graphics from iStockphoto[1] and other sources with permission.[2]

The Facebook version of the game was developed in PHP SDK (a Facebook API language allowing access to user data, friend lists, wall posting etc) and integrates seamlessly within the Facebook site. In order to play the game a Facebook user must grant certain permissions: the basic access (user details and friends list), which is required for all applications, and access to posting on the user's wall. Once the user has allowed the game access they never need to login to the game, only to Facebook.

[1] `http://www.istockphoto.com`

[2] `http://www.pixeljoint.com/p/3794.htm`, `http://p.yusukekamiyamane.com`



(67) Bristol Stool Scale - Wikipedia

ID	Text	Skip	Rels	Comments
9739	stool	0	6	0

RelID	AnteID	RelType	Annotations	Agree	Disagree	Total
7551	9746	DO	13	3	1	15
12227	9749	DO	2	1	3	0
15658		PR	3	0	4	-1
19661		DN	5	1	3	3
88682	9745	DO	2	0	4	-2
91261	9761	DO	2	0	4	-2

Figure C.1: Screenshot of the administrative tool to view markable statistics.

Administrative and analysis tools Several administrative tools were developed to analyse the data produced by the players and to manage inputs, outputs and users of the system:

- Analysis of game statistics. A selection of up-to-date statistics about the game that were useful for monitoring overall performance, such as total number of users, total words in the corpus, total annotations, average annotation times, throughput as well as ways of monitoring how these numbers change over time.
- Analysis of markable statistics. All annotations in the system broken down by document, then by markable, including annotations and validations for all interpretations, comments, skips and markables excluded from the system by administrators (see Figure C.1).
- Markable administration. All markables could be edited to correct mistakes created by the pre-processing pipeline or excluded from the game (but not deleted in order to maintain data integrity, see Figure C.2).
- Gold standard creation. An interface for experts to annotate documents (see Appendix D).
- Document management. All documents that were imported to the game could have metadata attached including complexity, language and whether the theme was of an adult nature.

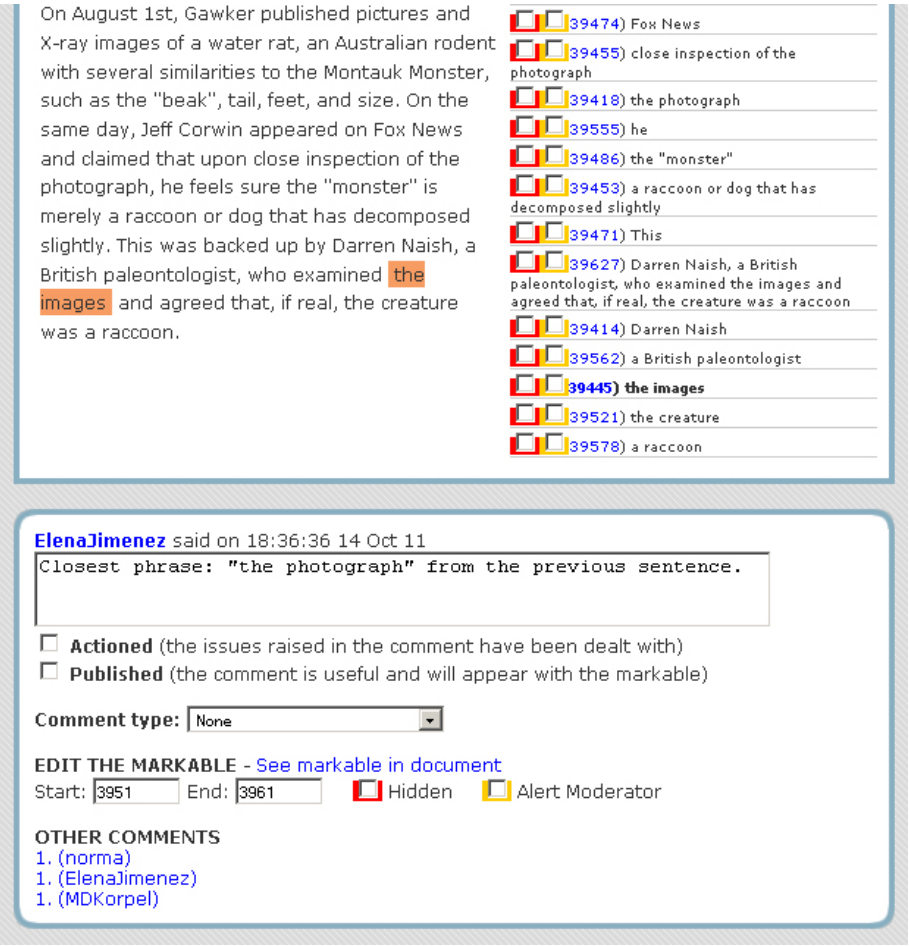


Figure C.2: Screenshot of the administrative tool to edit comments and markables.

- Comment management. Users were allowed to provide comments, and such comments have proven invaluable to identify problems with the pre-processing and the annotation scheme. All user comments could be viewed for a given markable and dealt with (see Figure C.2).

Markup of the *Phrase Detectives* corpora The markup used for the documents in the *Phrase Detectives* corpora was Minimum Anaphoric Syntax (MASXML) format [Kabadjov, 2007]. MASXML is a form of inline XML in which the basic information required to carry out entity references is marked, including:

- sentences;

- words with their part-of-speech tags (for English, the Penn Treebank tagset is used);
- NPs (called Nominal Entities, `ne`), with their ID and the basic agreement features: gender (attribute `gen` for gold-standard info, `AAgen` for automatically extracted information), number (again two attributes are used, `num` and `AAnum`), and person (using the attributes `per` and `AAper`);
- NP modifiers and heads, using the elements `mod` and `nphead`.

Note that the format does not require full syntactic information or Named Entity types. As an example, the representation in MASXML of the noun phrase *four little rabbits* is as follows:

```
<ne id="ne14819" AAcat="num-np"
    AAgen="neut" AAnum="plur" AAper="per3">
  <mod id="AAm2" AAcat="AApre">
     <W Lpos="CD">four</W>
     <W Lpos="JJ">little</W>
  </mod>
  <nphead id="AAh4">
     <W Lpos="NNS">rabbits</W>
  </nphead>
</ne>
```

Anaphoric information is marked using separate `ante` elements, a structured representation inspired by the Text Encoding Initiative `link` elements and that makes it possible to specify multiple anaphoric relations for each markable (identity and association) and to mark ambiguity using multiple `anchor` elements [Poesio, 2004b], as in the following (made-up) example:

```
<ante current="ne3" rel="identity">
   <anchor antecedent="ne1"/>
   <anchor antecedent="ne2"/>
</ante>
```

Pre-processing A text processing pipeline was developed to convert documents into the format importable in the database. The English pipeline that converted raw text to SGF format was developed by combining existing tools with ad-hoc modules for correcting the output of such tools in the case of frequent errors, as follows:

- A pre-processing step normalised the input, applied a sentence splitter and ran a tokeniser over each sentence. The tokeniser and sentence splitter used to perform this process were from the *openNLP* toolkit.[1]
- A custom-developed post-processing step was carried out to clean systematic errors by the tokeniser and sentence splitter.
- Each sentence was then analysed by the Berkeley Parser [Petrov et al., 2006].
- The parser output was then used to identify markables in the sentence. As a result a MASXML-like representation was created which preserved the syntactic structure of the markables (including nested markables, e.g. noun phrases within a larger noun phrase).
- A heuristic processor identified additional features associated with markables such as person, case, number, etc. The output format was MASXML.
- MASXML was converted to SGF using XSL stylesheets and Saxon.[2]

[1] `http://incubator.apache.org/opennlp`
[2] `http://saxon.sourceforge.net`

Appendix D

Creation of the *Phrase Detectives* gold standard

The gold standard was annotated through the *Phrase Detectives* administration system (see Figure D.1). The expert annotator was shown a list of all markable interpretations that have been entered by the players for a particular markable and could view each interpretation as if in Validation Mode in the text. By default the expert could not see how many annotations or validations each relation had scored. The markables were annotated in order of appearance in the text.

The expert then selected the best relation for the markable (the 'favoured' radio button) and selected the checkbox of any possible interpretations due to ambiguity. Additionally, if the markable was referring, the expert selected the checkboxes of any other relation that was the same entity. If the best relation was not mentioned in the list from the players, the expert could annotate the best markable relation as 'Not mentioned'. Markables that were ignored do not require an expert annotation.

Complete instructions and examples for experts on how to annotate apposition, discourse diexsis, out-of-context errors, questions, names, compound entities, bridging entities, temporal revelations, numerators and dates are detailed in Appendix E.

Of the 12 markables on which the experts did not agree in the *Phrase Detectives* W2 and G2 corpora, only one was an actual error in which the entity had been correctly identified but not the closest mention. The remaining markables fall into four categories of ambiguity:

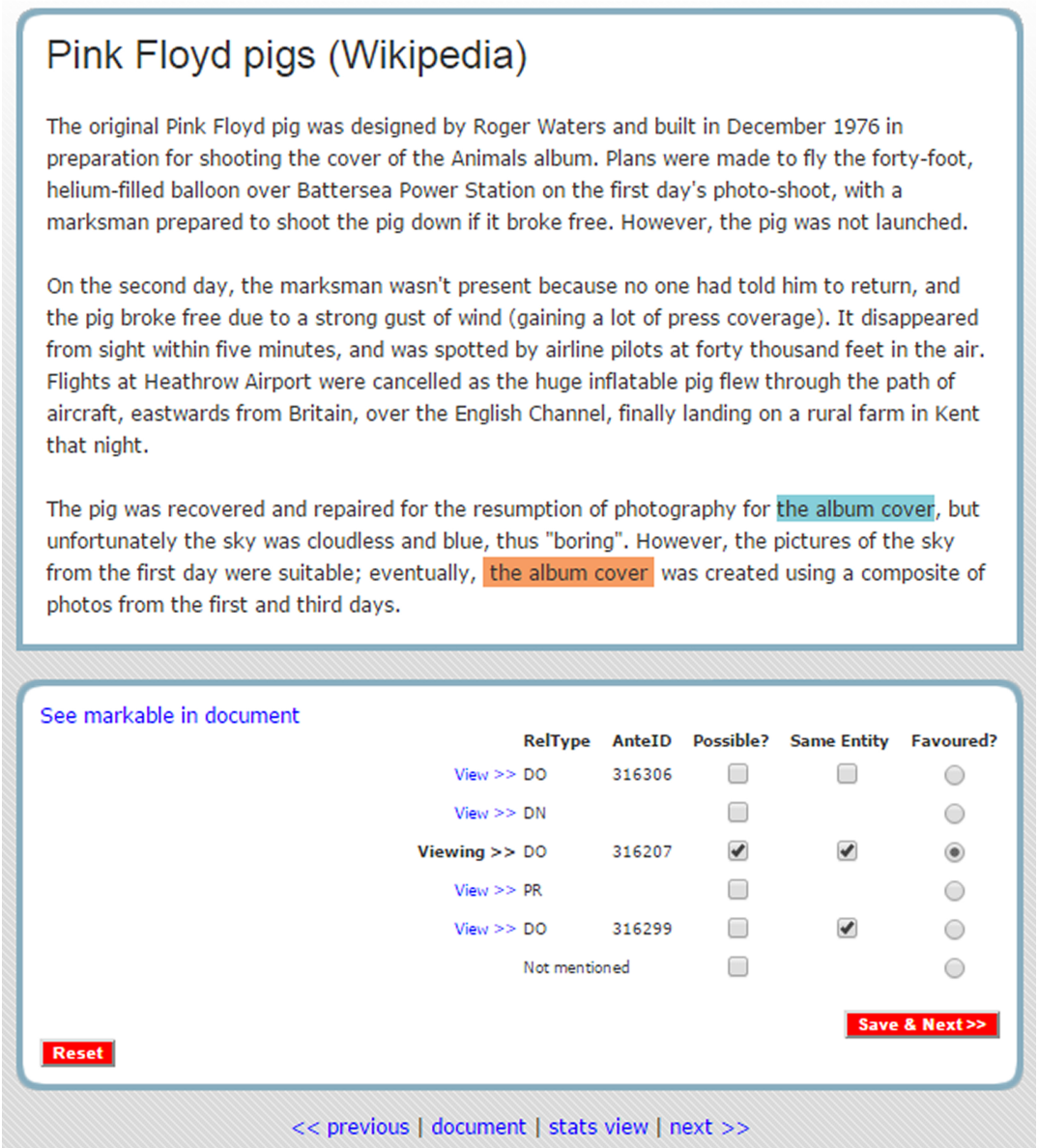


Figure D.1: The expert annotation administration interface for *Phrase Detectives*.

The first category of ambiguity is the **specificity** of the antecedent. The pre-processing chunks markables in a way that different levels of specificity can be selected, for example, in *Henry the Hexapus (Wikipedia)* the markable *the Blackpool Centre* refers to an earlier mention in the text of *the Blackpool Sea Life Centre in North West England*; however, a less specific markable within this markable is also selectable *the Blackpool Sea Life Centre*. Both interpretations are correct; however, clearer instructions would ensure this is marked up consistently.

The second category of ambiguity relates to **assumptions** the reader makes regarding the role of entities, for example in *Gay Fuel (Wikipedia)* it could be assumed that the acronym *LLC* and *Its maker* refer to the manufacturer of the drink *Florida-based Speciality Spirits*; however, this is not explicitly stated in the sentence 'Gay Fuel was an energy drink marketed by Florida-based Speciality Spirits' and the reader makes an assumption of the role of the entity. Prior knowledge of the reader is an important factor in the way they understand text and the context that is presented.

The third category of ambiguity is confusion over what constitutes a **property** of another markable and what is in fact another entity, for example, in *Gay Fuel (Wikipedia)* whether *bright pink and elderberry flavored* is a property of *the liquid* in '...the liquid was dyed bright pink and elderberry flavored.' Other examples of this type of ambiguity could explain why property markables are more difficult to annotate.

The final category of ambiguity is that of entity **generalisations**, in which perhaps coreference is not an appropriate annotation. For example, in *Human Mail (Wikipedia)* the mentions of *a person* in the sentences 'Human mail is the transportation of a person through the postal system.' and '...is the mailing of a part of a person...'

There were four markables without consensus in the GNOME corpus: the first was caused by GNOME using a slightly different annotation scheme for marking up the pronoun *who* as entities, found in *Cartonnier*; the second was caused by specificity of the entity (*the king at the age of twenty-one* being different to *the king*) found in *Cabinet on Stand*; and the last two markables, also found in *Cabinet on Stand*, lacked consensus because the assumption that *The Sun King* is the same entity as *King Louis XIV* was not explicitly stated in the text.

D. CREATION OF THE *PHRASE DETECTIVES* GOLD STANDARD

Appendix E

Instructions for creating the *Phrase Detectives* gold standard

As an expert using the Document Checking tool you will be presented with each markable from the document together with a list of interpretations that the players of *Phrase Detectives* have made. By clicking on the links for each interpretation the text will reform to highlight antecedent markables.

Choose the best interpretation for the markable (by clicking the *favoured* radio button) and select the checkbox of any possible interpretations due to ambiguity. Additionally, if the markable is referring, select the checkboxes of any other interpretation that is the same entity.

Common actions

- The correct interpretation not shown in the list – mark as *Not mentioned* favoured.
- Discourse diexsis – do not select anything (because a player could not select a correct answer).
- Markable error – edit or delete the markable as appropriate.
- The markable is not a noun phrase – do not select anything.
- The antecedent has been mentioned before, but is no longer visible in the shown text segment (known as an *out of context* error) – mark as *DN favoured*, with *Not mentioned* as a possible interpretation.

- The closest markable is not available for selection (this often happens with apposition, see below) – select nothing.

Due to the pre-processing of the documents a number of common linguistic features cause annotation errors and ambiguity. Below are examples of these errrors and how to annotate them in the *Phrase Detectives* scheme:

Apposition This is when a markable is embedded in a longer markable, but the embedded mention is not required in this scheme, for example the markable 'the 3rd king of England' being embedded in the markable 'Jon, the 3rd king of England'. In these cases use the head (longest) markable and do not select anything for the inner markables if they are the same entity. In this case 'the 3rd king of England' should be ignored, but 'England' should be annotated (as it is a separate entity). These markables are not deleted in the hope that future post-processing can make use of the annotations made on them.

An exception to this is when the head markable is embedded in the longer mention, for example the markable 'a great guy, Dave'. In this case annotate both markables.

Properties Properties can be specific or generalistic as in the case of *a clever person* in 'Jon is a clever person' and *the most clever person* in 'Jon is the most clever person'. Negative properties should be marked as DN, for example *a popular item* in 'It was never a popular item.' Properties that are similarities should be marked as DN, for example, *an old mop* in 'His beard was like an old mop.'

Questions The entity implied in a question cannot be referred and should be marked as DN, for example *Who* in 'Who is clever? Jon is'. However, following entities can refer to the entity implied in the question, such as *Jon*.

Names and naming The name of an entity is treated as a separate entity to the entity itself, for example, *Little Red Cap* does not refer to *She* in 'She was called Little Red Cap'.

Compound entities, joining and separating Compound entities can be very complicated to decipher. In simple cases the individual entities that are joined to create a compound entity such as the first *they* in 'Jon and Jim went to the pub, they got drunk. Jon had beer, Jim had whisky and they left.' Once the compound entity has been created it is preferred over rejoining individual entities, so in the above example the second *they* would refer to the first *they* and not to another joining of *Jon* and *Jim*. Mark joining interpretations as *possible* in this case.

When entities separate from a compound they are marked as DN, i.e. this is the first time they have been mentioned as an individual entity.

Great care must be taken when considering plural and compound entities as they may not refer to an entire group of entities as mentioned before. Each grouping of entities is marked as DN.

Collective references to individuals in a group refer to the group itself, for example, the markables *The suitors* and *each one of the suitors* mean the same thing.

Bridging relations Bridging relations are marked as DN such as *a door* in 'The house with a door.' or *the resutls* in 'There were several examinations and the results were good.'

Temporal revelations Revelations that are made during the context of the document should be marked up at the point of revelation and not retrospectively marked throughout the text, for example, do not mark *the butler* being the murderer through the detective novel when it is revealed at the end. Errors of this type may also be caused by prior knowledge of the annotators so try to annotate within the context and revelations presented in the text.

Numbers and measurements Numbers and measurements are usually marked as DN and are not referential to each other, but can be referred to, for example, *This number* refers to *12* in 'The number of pints he drank was 12. This number was too much.'.

Measurements can in some cases be properties, for example, *12m* is a property of *the length of the bridge* in 'The length of the bridge was 12m.'; however, *12m* would be marked as DN in 'The bridge was 12m in length.'

E. INSTRUCTIONS FOR CREATING THE *PHRASE DETECTIVES* GOLD STANDARD

Dates Only the whole date should be a markable (e.g. 7 November 2014). 7, November and 2014 are DN and only refer to each other as generic entities. Today, tomorrow, yesterday, etc., refer to a particular date entity (e.g. 7 November 2014) and can refer. Dates may also be properties of entities, for example, *7 November 2014* is a property of *the date* in 'The date was 7 November 2014'.

Entity confusion In some unusual and ambiguous cases entities can become confused in the text in which case the context of the document must be used to provide the best answer and all possible other ambiguous interpretations also marked.

Appendix F

Analysis of the *Phrase Detectives* gold standard corpora

Each gold standard dataset (see Table 5.1) was analysed for syntactic and structural differences (see Table F.1):

- The number of words per sentence (*W/S*) as calculated by PHP word count (str_word_count) on sentences chunked by the pre-processing.
- The number of words per active (not deleted) markable (*W/M*); the total number of words in each sentence (as calculated by a PHP word count of the content of the sentence) divided by the total active markables per sentence (as calculated by the pre-processing with deleted ones removed). Sentences with no active markables were ignored.
- The average (mean) proportion of markables that were deleted (*%M del*) or edited (*%M edit*) per document.
- The average (mean) readability of each document's content as calculated by an online assessment[1] of the Flesch Reading Ease Score (FRES) [Kincaid et al., 1975]. The score is calculated as weighted averages of words per sentence and syllables per word:

$$\text{FRES} = 206.835 - 1.015\frac{total\_words}{total\_sentences} - 84.6\frac{total\_syllables}{total\_words}$$

F. ANALYSIS OF THE *PHRASE DETECTIVES* GOLD STANDARD CORPORA

Table F.1: Summary analysis of gold standard datasets showing words per sentence (*W/S*), active markables per word (*M/W*), the proportion of markables that were deleted (*%M del*) and edited (*%M edit*) and the average (mean) Flesch Reading Ease score.

Corpus	*W/S*	*W/M*	*%M del*	*%M edit*	Readability
GN	19.4 sd(7.3) n(45)	3.4 sd(1.0) n(45)	0	0	52.3 sd(10.7) n(5)
W2	16.5 sd(7.2) n(30)	2.9 sd(0.6) n(29)	4.3	4.3	53.6 sd(5.6) n(5)
G2	18.0 sd(8.1) n(10)	3.2 sd(1.1) n(10)	7.2	0	88.2 n(1)
WL1	21.6 sd(9.7) n(303)	3.5 sd(0.9) n(301)	4.0	9.3	52.8 sd(8.4) n(8)
WS1	20.9 sd(10.2) n(289)	3.5 sd(1.0) n(285)	3.7	8.9	49.9 sd(11.2) n(22)
W1	21.3 sd(10.0) n(592)	3.5 sd(1.0) n(586)	3.9	9.1	50.7 sd(10.4) n(30)
G1	25.0 sd(17.9) n(249)	3.5 sd(1.0) n(248)	3.9	4.5	84.3 sd(3.5) n(4)

There is no significant difference between the short and long Wikipedia corpora when comparing:

- words per sentence (WS1 n(289) 20.9 sd(10.2); WL1 n(303) 21.6 sd(9.7); unpaired t-test, p=0.40)
- words per markable (WS1 n(285) 3.5 sd(1.0); WL1 n(301) 3.5 sd(0.9); unpaired t-test, p=0.93)
- markables deleted (WS1 n(1,898) 0.037; WL1 n(2,055) 0.040; z-test, p=0.62)
- markables edited (WS1 n(1,898) 0.089; WL1 n(2,055) 0.093; z-test, p=0.66)
- readability (WS1 n(22) 49.9 sd(11.2); WL1 n(8) 52.8 sd(8.4); unpaired t-test, p=0.51)

Looking at the results, G1 has a significantly longer average sentence length than W1 (G1 n(249) 25.0 SD(17.9); W1 n(592) 21.3 SD(10.0); unpaired t-test, p<0.01) but conversely is significantly easier to read (G1 n(4) 84.3 sd(3.5); W1 n(30) 50.7 sd(10.4); unpaired t-test, p<0.01). They show a similar number of words per markable of 3.5 sd(1.0).

There was no difference between the proportion of markables with errors that needed deleting found in the corpora (G1 n(1,971) 0.039; W1 n(3,953) 0.039; z-test, p=1); however, W1 required more markables per document to be edited (G1 n(1,971) 0.045; W1 n(3,953) 0.091; z-test, p<0.01), perhaps another indication that Wikipedia texts are harder to read and therefore harder to process.

[1] `https://readability-score.com`

GNOME shows no significant difference between Wikipedia perhaps an indication that these texts (mainly museum explanatory texts) are most similar to the Wikipedia encyclopedia entries:

- sentence length (GN n(45) 19.4 sd(7.3); W1 n(592) 21.3 sd(10.0); unpaired t-test, p=0.22)
- readability (GN n(5) 52.3 sd(10.7); W1 n(30) 50.7 sd(10.4); unpaired t-test, p=0.75),

Whilst there is a difference between sentence length and readability of W2 and G2 the corpora are too small to draw any firm conclusions from and they are mainly used for inter-expert agreement assessment.

It might be reasonable to assume that documents that are easier to read are also easier to process using automatic parsing; however, readability in this context only weakly correlates to the proportion of markables deleted (G1 and W1 n(34) R=0.16 R^2=0.024; Pearson, weak positive correlation) and edited (G1 and W1 n(34) R=0.28 R^2=0.077; Pearson, weak positive correlation).

F. ANALYSIS OF THE *PHRASE DETECTIVES* GOLD STANDARD CORPORA

Appendix G

Accessing and archiving data from social networks

In order to analyse the problem solving capabilities of social networks a pipeline to cache messages from Facebook groups was written in PHP and JavaScript and deployed on a live server. The software makes a request for a group's messages via the Facebook Graph API.[1] The call specifies the maximum number of messages to return (in date order, newest first) and the API returns a JSON encoded list of messages and metadata, termed here a **corpus**. The corpus is stored in JSON format in a MySQL database along with data about the group, such as the owner, title, description and privacy settings.

Each corpus contains a pagination link that is used to call sets of messages from a group. Pagination is used to minimise server load in processing large groups (avoiding timeout issues) and to circumvent Facebook's maximum message per call limit (500 messages). The software iterates through a group's messages from the latest message to the first message ever posted. The process of storing corpora from a group is termed here a **capture**.

The Facebook API was also used to find the gender of the each user, although users do not have to declare a gender or be truthful in their declaration, and their locale.

[1] `https://developers.facebook.com/docs/graph-api`

G. ACCESSING AND ARCHIVING DATA FROM SOCIAL NETWORKS

Table G.1: Categorised Facebook groups used for the groupsourcing image classification analysis.

Category and Group Name	Messages	Members	Facebook ID
Task Request - General (TR-G)			
ID Please (Marine Creature Identification)	700	990	396180553763159
Seasearch Identifications	702	298	341487989207852
	1,402		
Task Request - Specific (TR-S)			
British Marine Mollusca	96	75	119847231532929
Crustacean Identification Group	53	93	495449120535459
Echinoderms of the NE Atlantic	61	104	288717931183533
NE Atlantic Bryozoa	83	109	133129276808670
NE Atlantic Cnidaria	216	128	224626804295339
NE Atlantic Nudibranchs	977	348	166655096779112
NE Atlantic Tunicata	219	102	248476708561508
Nudibase - sharing Nudibranch knowledge	3,243	1,594	206426176075326
	4,948		
Media Gallery - General (MG-G)			
BSoUP Facebook Group	1,048	919	15647538540
Underwater Macro Photographers	4,607	8,248	166086283477622
UW photo - Fotosub	1,954	861	141274729364653
Wetpixel Underwater Photography	8,020	5,573	2212386016
	15,629		
Media Gallery - Specific (MG-S)			
Frogfish images	256	255	295624943879942
NUDIBRANCH LOVERS	1,395	799	209993015706410
Nudibranquios	551	222	49476188153
	2,202		
Knowledge Sharing - General (KS-G)			
British Marine Life Study Society	1,149	485	11262929875012
Marine Conservation Society (uk) SouthEast	268	294	47270594182
MarLIN	85	253	16755412655
National Forum for Biological Recording	87	178	239682369506506
Seasearch	137	415	2390232162
Seasearch East	370	185	271321002878334
Seasearch North East England	269	181	305151302854292
Seasearch North Wales	347	157	193611807335249
Seasearch Northwest England	222	66	136567993120179
Seaweed East 11	75	60	103293629771091
The Global Diving Community	4,911	3,496	416853248435154
The Tank Bangers	5,779	7,376	122110314530441
	13,699		
Knowledge Sharing - Specific (KS-S)			
AMPHIPODA	241	177	238356639577927
Crustacea of the NE Atlantic & NW Europe	89	87	407910645934570
EPAM Nudibranchs	324	194	374656695905614
Marine Flatworms	392	227	450219478324695
Porifera	113	174	319188528116865
	1,159		

Appendix H

Creation of the groupsourcing gold standard

The creation of the groupsourcing gold standard involved a number of manual analysis steps (all performed by Jon Chamberlain):

1. Each thread was manually analysed to extract every named entity (or interpretation to the image classification task) which were then normalised to a marine species ontology.[1] For example, when the text 'I think this is Chromodoris magnifica.' was analysed, the entity *Chromodoris magnifica* was extracted and assigned an ID of 558230, which was the unique identifier for that species in the ontology. Genus (more general than species level) classifications were ignored because the process of classifying an image to this level would be different, involving feature identification across morphological variations.

2. The thread sentiment was recorded for each named entity including positive and negative opinions and how many people liked the post. For example, when the text 'I think this is Chromodoris magnifica.' was analysed, a positive sentiment for the entity *Chromodoris magnifica* was recorded. Conversely, a negative sentiment was recorded for the text 'This is not Chromodoris magnifica.' Opinions from the same person were normalised, but likes were recorded as totals.

3. The highest-rated named entity for an image (totalling messages, replies and likes to replies) was presented to an expert annotator at random with the associated

[1] `http://www.marinespecies.org`

Figure H.1: The expert annotation interface for the social network gold standard.

thread image. This was checked using a variety of resources including three identification websites[1] [2] [3], Wikipedia, Encyclopedia of Life[4], an Android app[5] and books [Debelius and Peyer, 2004; Picton and Morrow, 1994] relevant to the geographical range of the group. Synonyms were also checked when it was difficult to find a match. When the image matched a classification on a resource a 'yes' was recorded; when the image for the species did not match the classification a 'no' was recorded. If the image could not be classified using that resource then no response was entered (see Figure H.1).

4. The classification was considered correct if the image was confirmed by the majority of the resources with the species name. The classification was not marked if it could not be found in any of the resources (as it could be a new name not updated to the resources) or if there was a split vote between the top-rated answer.

The methodology was biased from using only one expert annotator; however, further experts could not be used as they were few in number and involved in the responses that were being analysed here.

[1] `http://www.seaslugforum.net` (Bill Rudman)
[2] `http://www.nudibranch.org` (Jim Anderson)
[3] `http://www.medslugs.de` (Erwin Köhler)
[4] `http://www.eol.org`
[5] `http://www.inudibranch.com` (Gary Cobb)